



Universidad  
Carlos III de Madrid

**Departamento de Informática**

# **CONSTRUCCIÓN DE UNA HERRAMIENTA CASE DE SOFTWARE LIBRE PARA AYUDA AL DISEÑO SOFTWARE.**

## **SU USO EN UN CASO PRÁCTICO.**

**Autor:** Francisco Javier González Montalvo

**Tutor:** Luis García Sánchez

Leganés, Diciembre de 2011

## Agradecimientos

He recibido el apoyo de muchas personas a lo largo del desarrollo del presente proyecto final de carrera, pero sobre todo quiero agradecer la colaboración por su gran interés y apoyo, a mi tutor Luis García Sánchez, profesor de la Universidad Carlos III de Madrid.

También quiero agradecer a mi familia y pareja, por su incansable apoyo tanto para la realización de este proyecto, como en el resto de la carrera.

## Resumen

Este proyecto fin de carrera nos muestra como resultado del mismo un nuevo producto software, se presenta en forma de herramienta CASE y sirve de apoyo para la generación de diagramas de flujo de datos siguiendo la metodología Métrica versión 3. La técnica de los DFD, sirve para representar gráficamente los límites de un sistema y su lógica de procesos, estableciendo las funciones que hay que desarrollar.

También se pretende dar una visión sobre lo que es una herramienta CASE, para que sirva, su utilización y sus beneficios, así como lo que es una metodología de desarrollo del software.

La aplicación se ofrece que como una herramienta de software libre, por lo que puede ser usado, copiado, estudiado, modificado y redistribuido libremente.

Esta aplicación, desarrollada para la uc3m, tendrá como nombre *CASE-FREE 2008*.

## Abstract

This Final Year Project has yielded a new software product which consists of a CASE tool that can be employed as an assistance in generating data flowcharts in accordance with the version 3 Metric method. The DFD technique has the purpose of graphically representing the limits of a system and the process logic, thus establishing the features that need to be developed.

Furthermore, we also aim to show a view of what a CASE tool is, what can it be applied to, how it is used and the advantages of using it; as well as what a software development methodology consists of.

The application is offered as free software, which enables the user to use, copy, analyze, modify and redistribute it without limitation.

This application, developed for the UC3M, will be called *CASE-FREE 2008*.

# ÍNDICE

ÍNDICE DE TABLAS .....	10
1. INTRODUCCIÓN.....	13
1.1. OBJETIVOS.....	13
1.3. MEDIOS UTILIZADOS .....	15
2. ESTADO DE LA CUESTIÓN.....	18
2.1. INTRODUCCIÓN A LAS HERRAMIENTAS CASE.....	18
2.1.1. <i>Qué son las herramientas CASE</i> .....	18
2.1.1.1. Ventajas de utilizar las herramientas CASE.....	19
2.1.1.2. Desventajas de utilizar las herramientas CASE .....	20
2.1.2. <i>Historia</i> .....	20
2.1.3. <i>Evolución de las herramientas CASE</i> .....	22
2.1.4. <i>Clasificación de herramientas</i> .....	22
2.2. HERRAMIENTAS CASE EN EL MERCADO ACTUAL.....	24
2.3. METODOLOGÍA DEL DESARROLLO DEL SOFTWARE .....	24
2.3.1. <i>Metodología Métrica versión 3</i> .....	26
2.3.2. <i>Diagrama de Flujo de Datos</i> .....	29
2.3.2.1. Notación Métrica versión 3.....	30
3. GESTIÓN DEL PROYECTO.....	35
3.1. PLANIFICACIÓN.....	35
3.1.1. <i>Comparativa entre la planificación inicial y la planificación final</i> .....	39
3.2. ORGANIZACIÓN.....	40
3.3. PRESUPUESTO DEL PROYECTO .....	41
4. TECNOLOGÍA UTILIZADA.....	45
4.1. EVOLUCIÓN DE C.....	45
4.2. HISTORIA DE C#.....	46
4.3. MICROSOFT .NET.....	46
4.4.1. <i>Compilador</i> .....	50
5. ANÁLISIS DEL SISTEMA.....	52
5.1. REQUISITOS DE USUARIO .....	52
5.1.1. <i>Requisitos funcionales</i> .....	52
5.1.2. <i>Requisitos no funcionales</i> .....	62
5.2. REQUISITOS DEL SISTEMA .....	64
5.3. INSTALACIÓN .....	64
6. EJEMPLO UTILIZANDO CASE-FREE 2008 .....	65
6.1. ENUNCIADO.....	65

6.2. DFD DE CONTEXTO .....	65
6.3. DFD DE SISTEMA .....	67
6.4. DFD 1 .....	68
7. MANUAL DE USUARIO.....	70
7.1. INICIAR SESIÓN .....	70
7.2. CONECTAR FIGURAS ENTRE SÍ.....	71
7.3. OPCIONES DE MENÚ .....	71
7.4. OPCIONES DE RATÓN .....	110
7.5. EXPLOSIONAR PROCESOS .....	121
7.6. NOMBRAR PROCESOS.....	121
7.7. OTRAS UTILIDADES.....	121
8. DISEÑO DEL SISTEMA .....	123
8.1. DIAGRAMA DE CLASES .....	123
8.2. DETALLE DE LAS CLASES.....	125
9. CONCLUSIONES Y LÍNEAS FUTURAS .....	127
9.1. CONCLUSIONES .....	127
9.2. LÍNEAS FUTURAS.....	128
10. GLOSARIO DE TÉRMINOS Y DEFINICIONES.....	129
A.....	129
B.....	129
C.....	129
D.....	130
E.....	130
F.....	130
G.....	131
H.....	131
I.....	131
L.....	132
M.....	132
O.....	132
R.....	133
S.....	133
T.....	134
U.....	134
V.....	134
W.....	134
X.....	135
11. REFERENCIAS .....	136

## Lista de figuras

Figura 1: Variaciones en el significado de CASE.....	19
Figura 2: Fases de la metodología Métrica v3 .....	28
Figura 3: Entidad externa.....	30
Figura 4: Entidad externa (2) .....	31
Figura 5: Proceso .....	31
Figura 6: Proceso primitivo.....	32
Figura 7: Almacén de datos .....	33
Figura 8: Almacén de datos (2).....	33
Figura 9: Flujo de datos .....	34
Figura 10: Flujo doble.....	34
Figura 11: Diagrama Gantt estimado (1) .....	36
Figura 12: Diagrama Gantt estimado (2) .....	37
Figura 13: Diagrama Gantt real (1).....	38
Figura 14: Diagrama Gantt real (2).....	39
Figura 15: Organigrama .....	40
Figura 16: Logotipo de Microsoft .net .....	47
Figura 17: Marco de trabajo de la plataforma .NET .....	49
Figura 18: Microsoft Visual Studio2008 .....	51

Figura 19: Diagrama de contexto.....	66
Figura 20: Diagrama de sistema.....	68
Figura 21: DFD 1 .....	69
Figura 22: Vista nada más entrar en la aplicación .....	70
Figura 23: Conexión de dos formas .....	71
Figura 24: Vista nuevo diagrama .....	72
Figura 25: Vista de la opción abrir.....	73
Figura 26: Vista de la opción Configurar página.....	78
Figura 27: Vista de la opción Guardar como .....	79
Figura 28: Vista de la opción Imprimir.....	81
Figura 29: Funcionamiento del portapapeles. ....	82
Figura 30: Vista de la opción Seleccionar todo .....	83
Figura 31: Vista de la opción Buscar .....	84
Figura 32: Vista de la opción Eliminar páginas.....	85
Figura 33: Vista de la opción Zoom.....	86
Figura 34: Vista de la opción Propiedades.....	87
Figura 35: Vista Definir datos de formas.....	89
Figura 36: Vista de las reglas .....	90
Figura 37: Vista de la paleta de formas.....	91
Figura 38: Vista Informe de flujos .....	92



Figura 39: Informe en formato HTML .....	95
Figura 40: Vista de la opción Texto.....	100
Figura 41: Vista de la opción Línea .....	101
Figura 42: Vista de la opción Relleno.....	103
Figura 43: Vista de la opción Sombra.....	104
Figura 44: Vista de la opción Exportar a base de datos .....	107
Figura 45: Informe de la opción Chequear .....	108
Figura 46: Informe de la opción Balancear.....	108
Figura 47: Vista de la opción Ventana.....	110
Figura 48: Vista de la opción Protección .....	112
Figura 49: Vista de la opción Comportamiento .....	116
Figura 50: Vista de la opción Capa.....	117
Figura 51: Vista de la opción Gráfico de datos.....	119
Figura 52: Vista de la opción Reordenar páginas .....	120
Figura 53: Vista de la opción avance y retroceso .....	122
Figura 54: Diagrama de clases .....	124

## Índice de tablas

Tabla 1: Evolución de las herramientas CASE .....	22
Tabla 2: Clasificación de las herramientas CASE .....	23
Tabla 3: Evolución de las metodologías .....	26
Tabla 4: Restricciones de los flujos de datos .....	34
Tabla 5: Costes de personal .....	41
Tabla 6: Costes por hora según categoría profesional .....	42
Tabla 7: Costes de equipo .....	42
Tabla 8: Otros costes directos .....	43
Tabla 9: Resumen de costes .....	43
Tabla 10: Presupuesto total .....	44
Tabla 11: Requisito funcional 1 .....	52
Tabla 12: Requisito funcional 2 .....	53
Tabla 13: Requisito funcional 3 .....	53
Tabla 14: Requisito funcional 4 .....	54
Tabla 15: Requisito funcional 5 .....	54
Tabla 16: Requisito funcional 6 .....	54
Tabla 17: Requisito funcional 7 .....	55
Tabla 18: Requisito funcional 8 .....	55

Tabla 19: Requisito funcional 9 .....	55
Tabla 20: Requisito funcional 10 .....	56
Tabla 21: Requisito funcional 11 .....	56
Tabla 22: Requisito funcional 12 .....	56
Tabla 23: Requisito funcional 13 .....	56
Tabla 24: Requisito funcional 14 .....	57
Tabla 25: Requisito funcional 15 .....	57
Tabla 26: Requisito funcional 16 .....	57
Tabla 27: Requisito funcional 17 .....	58
Tabla 28: Requisito funcional 18 .....	58
Tabla 29: Requisito funcional 19 .....	58
Tabla 30: Requisito funcional 20 .....	59
Tabla 31: Requisito funcional 21 .....	59
Tabla 32: Requisito funcional 22 .....	60
Tabla 33: Requisito funcional 23 .....	60
Tabla 34: Requisito funcional 24 .....	60
Tabla 35: Requisito funcional 25 .....	61
Tabla 36: Requisito funcional 26 .....	61
Tabla 37: Requisito funcional 27 .....	61
Tabla 38: Requisito funcional 28 .....	62

Tabla 39: Requisito funcional 30 .....	62
Tabla 40: Requisito no funcional 1 .....	63
Tabla 41: Requisito no funcional 2 .....	63
Tabla 42: Requisito no funcional 3 .....	63
Tabla 43: Condiciones para los informes.....	94

# 1. Introducción

En el proyecto fin de carrera que se muestra en este libro, lo que se ha construido es una aplicación gráfica bajo el entorno de desarrollo Microsoft Visual Studio 2008, y para el cual hemos utilizado el lenguaje de programación denominado como C # o C Sharp.

Lo que se pretende al construir esta aplicación, es poder utilizar la técnica del diagrama de flujo de datos, bajo una metodología de desarrollo, en este caso Métrica v3, para poder representar el funcionamiento de un sistema en un proyecto software, es decir, lo que se ha construido es lo que se denomina como herramienta CASE, la cual se ofrece como software libre.

Las herramientas CASE son aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software, reduciendo el costo de las mismas en términos de tiempo y también de dinero. Además, tienen una gran influencia en la calidad del resultado final. Estas herramientas pueden abarcar una o más fases del ciclo de vida, en nuestro caso, nos hemos centrado en la etapa de análisis, por lo que podemos denominar a nuestra herramienta como herramienta Tools-Case, que son las que soportan una etapa dentro del ciclo de vida.

El motivo de la creación de este software es debido a la gran importancia que hoy en día tiene esta faceta de la informática, y, aunque las herramientas existentes cumplen con las necesidades actuales del mercado, ésta pretende ser más completa y potente que sus predecesoras, con muchas más opciones para el usuario, más sencilla de manejar y además, en español, ya que cuesta encontrar herramientas CASE en nuestro idioma, debido a su gran escasez.

Otro motivo de la creación de este software, es dar una visión amplia del mundo de las herramientas CASE, y dar a conocer los grandes beneficios que tienen este tipo de herramientas.

Este proyecto fin de carrera no está basado en ningún otro proyecto previo, es decir, que no se ha reutilizado código ni de proyectos anteriores, ni de terceras partes. Se parte del supuesto de que se ha hecho un análisis previo y se da validez en términos económicos y estratégicos a la realización y ejecución de este proyecto.

## 1.1. Objetivos

El principal objetivo que resume el cometido de este proyecto fin de carrera puede definirse como: diseño y desarrollo de diversas funcionalidades, que sirvan de herramienta de apoyo a equipos de desarrollo de software para la construcción de diagramas de flujo de datos,

bajo la notación Métrica v3. Es decir, el objetivo final del proyecto ha sido la construcción de una herramienta CASE.

Para lograr este objetivo final, el primer paso y más importante de todos, es conseguir integrar Microsoft Visio dentro de nuestro entorno de desarrollo. Es algo esencial, pues será la base sobre la cual vamos a construir la herramienta.

Una vez que logremos la integración, el siguiente objetivo será el diseño y la construcción de la interfaz gráfica, así como desarrollar toda la funcionalidad necesaria para la construcción de los diagramas.

Una vez finalizados los pasos anteriores, habremos logrado construir nuestra herramienta, pero como hemos dicho anteriormente, hoy en día existen multitud de productos similares en el mercado, por lo que éste tendrá que ser lo más completo posible. Los puntos que van a marcar la diferenciación de esta herramienta, serán los siguientes:

➤ **Compatibilidad con bases de datos:** La herramienta, aparte de guardar toda la información en el propio diagrama que estemos generando, nos permitirá exportarla a bases de datos externas para su posterior tratamiento y mantenimiento. Son varias las bases de datos a las que podremos exportar, como Excel, Access, Oracle y también SQL Server. Este será uno de sus puntos más fuertes, ya que la hace muy atractiva para los equipos de desarrollo acostumbrados a trabajar con estas tecnologías.

➤ **Presentación:** Otro objetivo importante es la presentación de los diagramas. Aparte de que un diagrama esté correctamente construido, nos vamos a centrar en darle una apariencia lo más vistosa posible. Para ello será posible darle multitud de formatos a las figuras, tanto en los contornos, como en el relleno e incluso aplicarle sombras. Esto también se aplicará a todas las cadenas de texto que puedan aparecer, ya que también podremos aplicarle multitud de formatos.

➤ **Facilidad de uso:** La herramienta tendrá una interfaz atractiva a la vez que sencilla, será muy intuitiva y seguirá un esquema de presentación de sus opciones muy estándar. También dispondrá de muchas opciones de configuración para adaptar los diagramas a nuestra manera de trabajar. Destacar también que será una herramienta en español, lo que nos facilitará más aún si cabe las cosas.

## 1.2. Fases de desarrollo

Para construir esta herramienta, hemos seguido los siguientes puntos o fases:

1. Una vez tenemos clara la idea de lo que se quiere construir y previo estudio de que es totalmente viable, lo primero que se hace es la toma de requisitos, para saber exactamente como debe comportarse la herramienta y qué necesidades se quieren cubrir.
2. A continuación, en una segunda fase, lo que se ha hecho es un análisis de las arquitecturas que se proponen como solución, para su posterior elección. También hemos hecho lo propio con el software que hemos utilizado.
3. En esta tercera fase, se ha tratado todo el diseño del software, tanto de las interfaces como de los procesos.
4. En la siguiente fase, la cuarta, lo que se ha hecho es el desarrollo de la aplicación. Se ha llevado a cabo la codificación tanto de las interfaces como de los procesos mediante el lenguaje de programación C # y bajo el entorno de desarrollo Microsoft Visual Studio 2008. Para hacer esto, hemos tomado como base el producto Microsoft Visio. Visio es un software de creación de dibujos y diagramas que facilita la visualización, el análisis y la comunicación de información compleja. Entre otras cosas, nos proporciona una superficie de dibujo para mostrar las formas, así como la posibilidad de crear una plantilla de formas totalmente personalizada y adaptada a nuestras necesidades. Lo que se ha hecho con Visio es integrarlo en nuestro entorno de desarrollo, para así tener acceso a toda su funcionalidad mediante su modelo de objetos. Con esto hemos podido adaptar o modificar parte de esta funcionalidad y también por supuesto, añadir nueva, para así construir una aplicación exacta a lo especificado en los requisitos.
5. En esta quinta fase, se han realizado todo tipo de pruebas al software, depurando a su vez el código cada vez que se ha detectado un defecto en el funcionamiento.
6. Se da por finalizada la creación de la herramienta y se hace entrega del software.

## 1.3. Medios utilizados

El software utilizado para la realización del presente proyecto ha sido:

- Microsoft Visual Studio 2008.

- Microsoft Visio en sus versiones 2003 y 2007 para garantizar su funcionamiento en todas.
- Windows XP, Windows Vista y Windows 7 para garantizar su funcionamiento en todas las versiones.

El hardware utilizado ha sido:

- Ordenador COMPAQ, con un procesador ADM Athlon <sup>TM</sup> II X2 220 a 2.80 GHz, 4,00 GB de memoria RAM y 586 GB de disco duro.

Otros medios utilizados han sido la conexión a internet y libros para ayuda en la programación del código.

## 1.4. Estructura del proyecto

Este documento se ha estructurado en las siguientes secciones generales:

1. **Introducción:** En esta sección, se hace una breve introducción de lo que se muestra en este proyecto fin de carrera, para seguidamente dar paso a explicar los objetivos que lo motivan, las fases que hemos seguido para su construcción, así como los medios empleados tanto de software, hardware....
2. **Estado de la cuestión.** En esta sección se pretende dar una visión de lo que son las herramientas CASE. Se muestra qué son, sus ventajas y desventajas, una introducción histórica de cómo surgen, su evolución y su clasificación en función al ciclo de vida. También veremos algunas de las herramientas que existen en el mercado.

También en esta sección se explica qué es la metodología del desarrollo de software, y nos centraremos en Métrica v3, que es la utilizada en nuestra aplicación, explicando lo que es un diagrama de flujo de datos y mostrando su notación.

3. **Gestión del proyecto.** En esta sección se detallan los aspectos relacionados con la gestión del proyecto, entre los que destacamos como más importantes la planificación, la organización y el presupuesto del mismo.



4. **Tecnología utilizada.** En esta sección se explica la tecnología utilizada para el desarrollo de la aplicación. Para ello veremos cómo ha evolucionado el lenguaje de programación C y la historia de C#, así como el compilador que hemos utilizado para su desarrollo.
5. **Análisis del sistema.** En esta sección se hace un análisis global de todo el sistema. Se muestran los requisitos del usuario, tanto los funcionales como los no funcionales. Se explican también los requisitos mínimos del sistema para la instalación y como se lleva a cabo.
6. **Ejemplo utilizando CASE.** En esta sección se plantea un enunciado para desarrollar un sistema y se muestra la solución dada utilizando nuestra herramientas CASE.
7. **Manual de usuario.** Esta sección corresponde a un manual de usuario para la aplicación desarrollada, donde se explica al detalle todas las opciones que tiene la herramienta y cómo utilizarla.
8. **Diseño del sistema:** En esta sección se muestra el diagrama de clases del sistema y a continuación se hace una descripción de cada una de ellas, indicando cómo están compuestas y su funcionamiento dentro del sistema.
9. **Conclusiones y líneas futuras.** En esta sección del documento se muestran las conclusiones obtenidas tras la finalización del proyecto, así como las líneas futuras de trabajo, que tienen como fin ampliar las funcionalidades de la herramienta CASE desarrollada.
10. **Glosario de términos y definiciones.** En esta sección se enumera todo el glosario de términos y definiciones utilizadas en el documento, para facilitar su lectura y comprensión.
11. **Referencias:** En esta sección se enumeran las fuentes de información consultadas para la realización del proyecto, mostrando referencias bibliográficas y portales de internet. Al principio de los capítulos se hace referencia a aquellas que nos han ayudado a su desarrollo.

## 2. Estado de la cuestión

Esta sección refleja el punto de partida en el cual comienza el presente proyecto, cuyo objetivo es poder situar al lector en el mundo de las herramientas CASE y las metodologías del software, con el fin de comprender mejor los objetivos del proyecto. [HC] [1] [2] [11]

### 2.1. Introducción a las herramientas CASE

Hoy en día, muchas empresas se han extendido a la adquisición de herramientas CASE, con el fin de automatizar los aspectos clave en los procesos de desarrollo de un sistema, desde el principio hasta el final, y así incrementar su posición en el mercado competitivo.

En algunas ocasiones se obtienen elevados costes, tanto en la adquisición de la herramienta como en el entrenamiento de personal. La falta de adaptación de la herramienta a la arquitectura de la información y a las metodologías de desarrollo utilizadas por la propia empresa, también pueden ser un inconveniente.

Por otra parte, algunas herramientas CASE no ofrecen o evalúan soluciones potenciales para los problemas relacionados con sistemas y no llevan a cabo ningún análisis de los requerimientos de la aplicación.

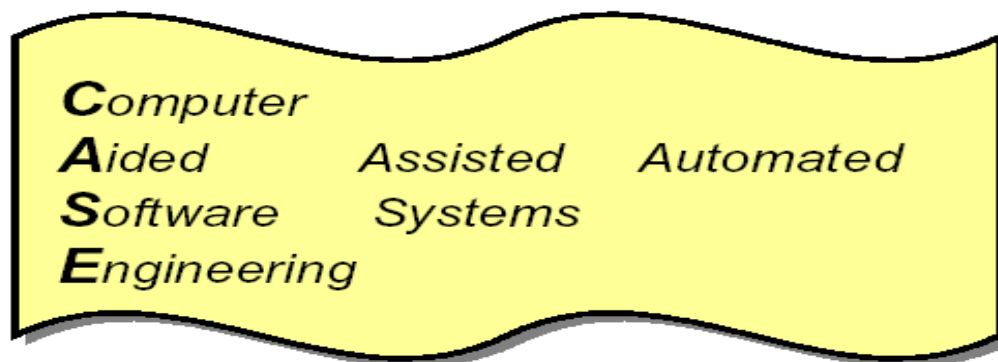
Sin embargo, CASE proporciona un conjunto de herramientas semiautomatizadas y automatizadas que están desarrollando una cultura de ingeniería nueva para muchas empresas. Uno de los objetivos más importante del CASE (a largo plazo) es conseguir la generación automática de programas desde una especificación a nivel de diseño.

#### 2.1.1. Qué son las herramientas CASE

Se puede definir a las herramientas CASE como un conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del ciclo de vida de desarrollo de un software.

Otras definiciones de herramientas CASE pueden ser:

- Conjunto de métodos, utilidades y técnicas que facilitan la automatización del ciclo de vida del desarrollo de sistemas de la información, completamente o en alguna de sus fases.
- La sigla genérica para una serie de programas y una filosofía de desarrollo de software, que ayuda a automatizar el ciclo de vida de desarrollo de los sistemas.
- Una innovación en la organización, un concepto avanzado en la evolución de tecnología como un potencial efecto profundo en la organización. Se puede ver al CASE como la unión de las herramientas automáticas de software y las metodologías de desarrollo de software formales.



*Figura 1: Variaciones en el significado de CASE*

#### **2.1.1.1. Ventajas de utilizar las herramientas CASE**

Algunas de las ventajas o beneficios de estas herramientas son:

- Facilidad para llevar la revisión de especificaciones del sistema, así como de representaciones gráficas.
- Facilidad para desarrollar prototipos de sistemas por medio de la capacidad para cambiar especificaciones y, por otro lado, determinar el efecto sobre el sistema que tendrían otras alternativas.
- Verificar el uso de todos los elementos en el sistema diseñado.
- Automatizar el dibujo de diagramas.
- Ayudar en la documentación del sistema.
- Ayudar en la creación de relaciones en la base de datos.
- Ayudar a la implantación de una metodología de trabajo, para todo el ciclo de vida.
- Generar estructuras de código.
- Comprobación de sintaxis y errores.

- Garantizar la consistencia de los procedimientos.
- Disminución de tiempo y reducción de costes, especialmente de mantenimiento.

### **2.1.1.2. Desventajas de utilizar las herramientas CASE**

Las herramientas case, también poseen sus debilidades, algunas de ellas son:

- Confiabilidad en los métodos estructurados: No todas las organizaciones utilizan métodos de análisis estructurado.
- Conflictos en el uso de los diagramas: Se producen entre las antiguas y las nuevas herramientas, así como por las herramientas visuales y no visuales.
- Tareas humanas: Esta actividad sigue siendo crítica, la experiencia es mayor que la máquina.
- Alcance limitado: No llevan a cabo ningún análisis de requerimientos de la aplicación. Tampoco garantiza que dos analistas que utilicen los mismos métodos aplicados de forma idéntica, hagan recomendaciones igualmente aceptables.
- Función limitada: Por lo general, su enfoque primario está dirigido hacia una fase o método específico.
- Diagramas no utilizados: Investigadores afirman, que en algunos casos, las herramientas gráficas no se emplean del todo, o, no como debiesen.
- Falta de niveles estándar para el soporte de la metodología: Una herramienta puede dar soporte a los diagramas que emplea una metodología o soportarlos e imponer la metodología. Las herramientas independientes de la metodología, por lo general, no pueden fomentar el uso de las reglas o estándares de la misma.
- Superación de puntos ciegos del desarrollador: quienes las desarrollan, pueden tener lagunas al comprender qué es lo que el analista o programador quieren exactamente.

### **2.1.2. Historia**

Las herramientas CASE tienen su inicio con un simple procesador de palabras, que fue usado para crear y manipular documentación. Más tarde se introdujeron técnicas gráficas y diagramas de flujo, aunque el diseño gráfico era extremadamente complejo y consumía mucho tiempo para realizar cambios.

La introducción de las herramientas CASE para ayudar en este proceso ha permitido que los diagramas puedan ser fácilmente creados y modificados, mejorando la calidad de los diseños de software. Los diccionarios de datos, un documento muy usado que mantiene los detalles de

cada tipo de dato y los procesos dentro de un sistema, son el resultado directo de la llegada del diseño de flujo de datos y análisis estructural, hecho posible a través de las mejoras en las herramientas CASE.

Pronto se reemplazaron los paquetes gráficos por paquetes especializados que habilitaban la edición, actualización e impresión en múltiples versiones de diseño. Eventualmente, las herramientas gráficas integradas con diccionarios de base de datos para producir poderosos diseños y desarrollar herramientas, podían sostener ciclos completos de diseño de documentos.

Como un paso final, la verificación de errores y generadores de casos de pruebas fueron incluidos para validar el diseño del software. Todos estos procesos pueden saberse integrados en una simple herramienta CASE que soporta todo el ciclo de desarrollo.

La primera herramienta comercial se remonta a 1982, aunque algunos especialistas indican que algunos ejemplos de herramientas para diagramación ya existían.

No fue sino hasta 1985 cuando las herramientas CASE se volvieron realmente importantes en el proceso de desarrollo de software. Los proveedores prometieron a la industria que muchas actividades serían beneficiadas por la ayuda de las CASE.

Estos beneficios consistían, por ejemplo, en el aumento en la productividad. El objetivo en 1985 para muchos vendedores era producir software más rápidamente.

Las herramientas del CASE serían una familia de métodos favorablemente estructurados para el análisis y diseño. Esto llevaría a la generación automática de código para desarrollo de software mediante una especificación formalmente diseñada. Esto traería como beneficio:

- ✚ Una mejora en la calidad, fiabilidad, utilidad y rendimiento.
- ✚ El entorno de producción de documentación para software mejora la comunicación, mantenimiento y actualización
- ✚ Hace el trabajo de diseño de software más fácil y agradable.
- ✚ La promesa futura de reemplazar realmente a los ingenieros de software especializados.
- ✚ Reducción del coste de producción de software.

Con estos objetivos en mente, la industria destinó millones en producción de herramientas CASE.

Así como esta enorme suma de dinero fue gastada en herramientas CASE, hubo también trabajo de investigación a nivel mundial en diferentes instituciones como universidades, instituciones gubernamentales y de defensa. La industria de herramientas CASE está creciendo y está tomando cada vez mayor importancia.

### 2.1.3. Evolución de las herramientas CASE

A continuación vemos una tabla con la evolución que han tenido las herramientas CASE desde inicio de los años ochenta.





<i>A inicios de los 80's</i>	<ul style="list-style-type: none"><li>- Ayuda en la documentación por ordenador.</li><li>- Diagramación asistida por ordenador.</li><li>- Herramientas de análisis y diseño.</li></ul>
<i>A mediados de los 80's</i>	<ul style="list-style-type: none"><li>- Diseño automático de análisis y pruebas.</li><li>- Repositorios automáticos de información de sistemas.</li></ul>
<i>A final de los 80's</i>	<ul style="list-style-type: none"><li>- Generación automática de código desde especificaciones de diseño.</li></ul>
<i>A inicio de los 90</i>	<ul style="list-style-type: none"><li>- Metodología inteligente.</li><li>- Interface de usuario reusable como una metodología de desarrollo</li></ul>

*Tabla 1: Evolución de las herramientas CASE*

### 2.1.4. Clasificación de herramientas

No existe una única clasificación de herramientas CASE y, en ocasiones, es difícil incluirlas en una clase determinada.

Podrían clasificarse atendiendo a:

-  Las plataformas que soportan.
-  Las fases del ciclo de vida que cubren.
-  La arquitectura de las aplicaciones que producen.
-  Su funcionalidad.

La siguiente clasificación es la más habitual y está basada en función de las clases del ciclo de vida que abarca:

1. Herramientas integradas, I-CASE: abarcan todas las fases del ciclo de vida del desarrollo de sistemas. Son llamadas también CASE workbench.
2. Herramientas de alto nivel, U-CASE o front-end, orientadas a la automatización y soporte de las actividades desarrolladas durante las primeras fases del desarrollo: análisis y diseño.
3. Herramientas de bajo nivel, L-CASE o back-end, dirigidas a las últimas fases del desarrollo: construcción e implantación.
4. Juegos de herramientas o Tools-Case, son el tipo más simple de herramientas CASE. Automatizan una fase dentro del ciclo de vida. Dentro de este grupo se encontrarían las herramientas de reingeniería, orientadas a la fase de mantenimiento.

A continuación, vemos un resumen de las ventajas y desventajas según esta última clasificación:

<i>Tipo de Case</i>	<i>Ventajas</i>	<i>Desventajas</i>
<b><i>I-CASE</i></b>	<ul style="list-style-type: none"><li>- Integra el ciclo de vida.</li><li>- Logra importantes mejoras de productividad a medio plazo.</li><li>- Permite un eficiente soporte al mantenimiento de sistemas.</li><li>- Mantiene la consistencia de los sistemas a nivel corporativo</li></ul>	<ul style="list-style-type: none"><li>- Es más eficiente para soluciones complejas que para simples.</li><li>- Depende del hardware y del software.</li><li>-Es costoso.</li></ul>
<b><i>U-CASE</i></b>	<ul style="list-style-type: none"><li>- Es aplicable a diversos entornos.</li><li>-Menos costo.</li></ul>	<ul style="list-style-type: none"><li>- Permite mejorar la calidad de los sistemas, pero no la productividad.</li><li>- No permite la integración del ciclo de vida</li></ul>
<b><i>L-CASE</i></b>	<ul style="list-style-type: none"><li>- Permite lograr importantes mejoras a corto plazo.</li><li>- Permite un eficiente soporte al mantenimiento de sistemas.</li></ul>	<ul style="list-style-type: none"><li>- No garantiza la consistencia de los resultados a nivel corporativo.</li><li>- No garantiza la eficiencia del diseño y en análisis.</li><li>- No permite la integración del ciclo de vida</li></ul>

*Tabla 2: Clasificación de las herramientas CASE*

## 2.2. Herramientas CASE en el mercado actual

En esta sección vamos a presentar de forma breve alguna de las herramientas CASE que existen en el mercado actual.

- **Erwin:** Es una herramienta de diseño de base de datos. Brinda productividad en diseño, generación, y mantenimiento de aplicaciones. Genera automáticamente las tablas y miles de líneas de stored procedure y triggers para los principales tipos de base de datos. Erwin hace fácil el diseño de una base de datos. Establece una conexión entre una base de datos diseñada y una base de datos, permitiendo transferencia entre ambas y la aplicación de ingeniería inversa. Usando esta conexión, Erwin genera automáticamente tablas, vistas, índices, reglas de integridad referencial (claves primarias, claves foráneas), valores por defecto y restricciones de campos y dominios.

- **PowerDesigner:** PowerDesigner es una suite de aplicaciones para la construcción, diseño y modelado de datos a través de diversas aplicaciones. Es la herramienta para el análisis, diseño inteligente y construcción sólida de una base de datos y un desarrollo orientado a modelos de datos a nivel físico y conceptual, que da a los desarrolladores Cliente/Servidor una base firme para aplicaciones de alto rendimiento.

- **System Architect:** System Architect posee un repositorio único que integra una amplia variedad de metodologías. En la elaboración de los diagramas, conecta directamente al diccionario de datos, los elementos asociados, comentarios, reglas de validaciones, normalización, etc. Posee el control automático de diagramas y datos, normalizaciones y balanceo entre diagramas. Posee también esquemas de seguridad e integridad a través de contraseñas que posibilitan el acceso al sistema en diversos niveles.

## 2.3. Metodología del desarrollo del software

Como concepto general, una metodología es una manera ordenada y sistemática de proceder para obtener algún fin. Para el ámbito de estudio de este proyecto, metodología será el conjunto de pasos y procedimientos que deben seguirse para el desarrollo y mantenimiento de un producto software. Una metodología de desarrollo de software incluirá, los procedimientos, técnicas, herramientas y un soporte documental que ayudarán a los desarrolladores a realizar nuevo software.

Normalmente una metodología consiste en un conjunto de fases descompuestas en subfases (módulos, etapas, pasos, etc.). Esta descomposición del proceso de desarrollo, guía a los desarrolladores en la elección de las técnicas que debe elegir para cada estado del proyecto, y



facilita la planificación, gestión, control y evaluación de los proyectos. Una metodología, por tanto, representa el camino para desarrollar software de una manera sistemática.

La descomposición del proceso se realiza hasta el nivel de tareas o actividades elementales. Para cada tarea se identifica un procedimiento que define la forma de ejecutarla y es el vehículo de comunicación entre usuarios y desarrolladores. Como resultado de seguir un procedimiento, se obtienen uno o más productos (éstos pueden ser productos intermedios, si sirven de base para realizar nuevos productos durante el desarrollo o productos finales).

Para aplicar un procedimiento se pueden utilizar una o más técnicas. Éstas suelen ser, con mucha frecuencia, gráficas con apoyos textuales formales, como los DFD de la metodología Métrica v3, y determinan el formato de los productos resultantes de cada tarea. Podemos decir que las técnicas se pueden utilizar varias veces dentro de una metodología o en metodologías diferentes.

Para la realización de una técnica, podemos apoyarnos en las herramientas CASE, que automatizan en mayor o en menor grado su aplicación. Algunas herramientas se han diseñado para que proporcionen soporte específico a una metodología y, por lo tanto, a las técnicas que necesita, pero también se pueden encontrar herramientas de propósito más general que dan soporte a varias metodologías.

En la siguiente tabla podemos ver cómo han surgido las metodologías más representativas en la historia de la Ingeniería del Software.

<b>Años</b>	<b>Metodologías</b>
<b>1968</b>	Concepción sobre la programación estructurada de DIJKSTRA.
<b>1974</b>	Técnicas de programación estructurada de WARNIER y JACKSON.
<b>1975</b>	Primeros conceptos sobre diseño estructurado de MYERS y YOURDON
<b>1977</b>	Primeros conceptos sobre el análisis estructurado de GANE Y SARSON.
<b>1978</b>	Análisis estructurado: DEMARCO Y WIINBERG. Nace MERISE.
<b>1981</b>	SSADM Versión inicial. Information Engineering Versión inicial.
<b>1985</b>	Análisis y diseño estructurado para sistemas de tiempo real de WARD y MELLOR.
<b>1986</b>	SSADM Versión 3.
<b>1987</b>	Análisis y diseño estructurado para sistemas de tiempo real e HATLEY y PIRHBAY.
<b>1989</b>	MÉTRICA Versión inicial .Creador: ERITEL.
<b>1990</b>	SSADM Versión 4.
<b>1993</b>	MÉTRICA Versión 2. Creador: Coopers & Lybrand.
<b>1995</b>	MÉTRICA Versión 2.1. Creador: Universidad Carlos III.
<b>1999</b>	Rational Unified Process (RUP).
<b>1999</b>	Extreme Programming (XP).
<b>2000</b>	MÉTRICA Versión 3. Creador: IECISA y CSI.
<b>2002</b>	Enterprise Unified Process (EUP) extensiones RUP.
<b>2004</b>	Constructionist design methodology (CDM) por Kristinn R. Thórisson.
<b>2005</b>	Agile Unified Process (AUP) por Scott Ambler.

*Tabla 3: Evolución de las metodologías*

### 2.3.1. Metodología Métrica versión 3

Métrica es una metodología de planificación, desarrollo y mantenimiento de los sistemas de información. Su primera versión se publicó en 1989 por Eritel y fue promovida por el Ministerio de Administraciones Públicas del gobierno para la sistematización de actividades del ciclo de vida de los proyectos software en el ámbito de las administraciones públicas. Desde entonces y hasta la actualidad se han publicado cuatro versiones diferentes y se han extendido hasta el resto de empresas e instituciones.

Métrica v3 tiene su punto de partida en la versión anterior, la 2.1, de la cual se han conservado la adaptabilidad, flexibilidad y sencillez, así como la estructura de actividades y tareas y permite alcanzar los siguientes objetivos:

- Proporcionar o definir sistemas de información que ayuden a conseguir los fines de la organización mediante la definición de un marco estratégico para el desarrollo de los mismos.
- Dotar a la organización de productos software que satisfagan las necesidades de los usuarios dando una mayor importancia al análisis de requisitos.
- Mejorar la productividad de los departamentos de sistemas y tecnologías de la información y las comunicaciones, permitiendo una mayor capacidad de adaptación a los cambios y teniendo en cuenta la reutilización en la medida de lo posible.
- Facilitar la comunicación y entendimiento entre los distintos participantes en la producción de software a lo largo del ciclo de vida del proyecto, teniendo en cuenta su papel y responsabilidad, así como las necesidades de todos y cada uno de ellos.
- Facilitar la operación, mantenimiento y uso de los productos software obtenidos.

Los procesos de su estructura principal son los siguientes:

1. **Planificación de sistemas de la información:** Su objetivo es proporcionar un marco estratégico de referencia en un ámbito determinado de la organización, para orientar las actuaciones en materia de desarrollo.
2. **Desarrollo de sistemas de la información:** Contiene todas las actividades y tareas que se deben llevar a cabo para desarrollar un sistema, cubriendo desde el análisis de requisitos hasta la instalación del software. Se divide en los siguientes cinco subprocesos:
  - Estudio de viabilidad del sistema (EVS).
  - Análisis del sistema de información (ASI).
  - Diseño del sistema de información (DSI).
  - Construcción del sistema de información (CSI).
  - Implantación y aceptación del sistema (IAS).
3. **Mantenimiento de sistemas de información:** Su objetivo es la obtención de una nueva versión de un sistema de información a partir de las peticiones de mantenimiento que realizan los usuarios bien por problemas o bien por mejoras.

Otro de los aportes de esta versión de métrica son las interfaces, que definen una serie de actividades de tipo organizativo o de soporte al proceso de desarrollo y a los productos, que en el caso de existir en la organización se deberán aplicar y sino, se deberán realizar para complementar y garantizar el éxito del proyecto. Esas interfaces son:

- **Gestión de Proyectos (GP):** Su finalidad principal es la planificación, el seguimiento y control de las actividades y de los recursos humanos y materiales que intervienen en el desarrollo de un sistema de información para conocer en todo momento que problemas se producen.
- **Seguridad (SEG):** Su objetivo es incorporar en los sistemas de información mecanismos de seguridad asegurando el desarrollo de cualquier tipo de sistema a lo largo de los procesos que se realicen para su obtención.
- **Gestión de la Configuración (GC):** Su finalidad es identificar, definir, proporcionar información y controlar los cambios en la configuración del sistema, así como las modificaciones y versiones de los mismos. Este proceso permitirá conocer el estado de cada uno de los productos que se hayan definido como elementos de configuración, garantizando que no se realizan cambios incontrolados y que todos los participantes en el desarrollo del sistema disponen de la versión adecuada de los productos que manejan.
- **Aseguramiento de la Calidad (CAL):** Su objetivo es proporcionar un marco común de referencia para la definición y puesta en marcha de planes específicos de aseguramiento de calidad aplicables a proyectos concretos.

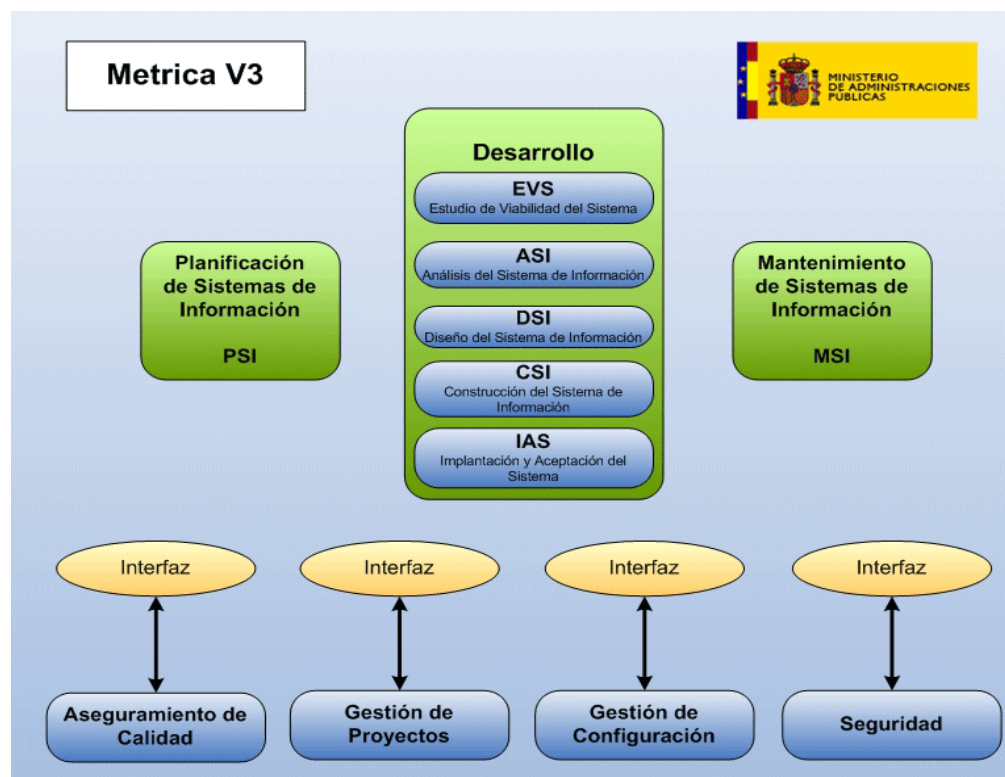


Figura 2: Fases de la metodología Métrica v3

### 2.3.2. Diagrama de Flujo de Datos

El diagrama de flujo de datos es una técnica del análisis estructurado para la representación de la función del software. Es el método más extendido del modelado conceptual de procesos.

Es una técnica gráfica que representa el flujo de la información que se mueve por el sistema y las transformaciones que se aplican a los datos al moverse desde la entrada hasta la salida.

La técnica del DFD se basa en el principio de descomposición de la funcionalidad de un sistema en sucesivos niveles de refinamiento (enfoque top-down), cuyo objetivo es permitir una lectura de la especificación del sistema desde lo abstracto al detalle, permitiendo ver el sistema de forma global (niveles superiores), o de forma detallada (niveles inferiores). Queda organizado así en una serie de niveles de modo que, cada uno proporcione sucesivamente más detalles sobre una porción del nivel anterior.

Así se parte de un diagrama general o diagrama de contexto, en el que se representa una sola función con el nombre del sistema, las entidades externas que se relacionan con el sistema y, los flujos de entrada y salida al sistema.

El siguiente nivel conocido como DFD/0 o diagrama de sistema, representa la vista de más alto nivel de las principales funciones del sistema, al igual que sus principales interfaces. Y así sucesivamente.

La descomposición de cada subsistema no se detiene en un nivel determinado, sino que dependen del grado de complejidad del sistema. Normalmente se recomienda no utilizar más de cuatro niveles y que en cada nivel no aparezcan más de nueve funciones.

Veamos ahora algunos aspectos destacados de cada nivel:

**Diagrama de contexto (Nivel inicial).** El objetivo de este diagrama es la delimitación del dominio del sistema que se está estudiando. Formado por una única función, que representa al sistema completo y el contexto del mismo.

**Diagrama de Sistema.** Se conoce como diagrama de nivel uno, aunque también como diagrama cero, porque representa la explosión del nivel cero, que es el diagrama de contexto. En él se representan las funciones principales. Se representarán procesos que sean conceptualmente independientes entre sí para poder dividir el trabajo.

**Diagramas de niveles intermedios.** Corresponden a los DFD en los que se explotan o descomponen cada uno de los subsistemas.

**Diagramas de último nivel.** Son aquellos en los que las funciones que aparecen no se van a descomponer en subfunciones. Suelen tener pocos flujo de entrada y salida y además bastante sencillos.

### 2.3.2.1. Notación Métrica versión 3

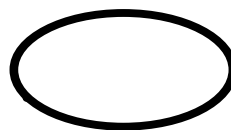
Los componentes de un DFD siguiendo la notación Métrica v3 son los siguientes:

**Entidad externa:** Representan la fuente y/o el destino de la información del sistema y puede ser una persona, grupo de personas o una unidad de negocio.

Mediante la designación de alguna cosa o de algún sistema como entidad externa, estamos estableciendo implícitamente que se encuentra fuera de los límites del sistema que estamos considerando, por lo cual, no nos interesa la transformación o proceso que se realiza dentro de ellos, es decir, que están fuera del control del sistema que se está modelando. Por todo ello, ni el analista ni el diseñador pueden cambiar ni los contenidos ni la forma de trabajo de una entidad externa.

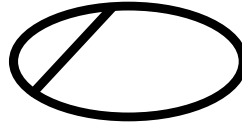
Algunas características de las entidades externas son las siguientes:

- Representa personas, organizaciones, o sistemas que no pertenecen al sistema.
- En el caso de que las entidades externas se comunicasen entre sí, esto no se contemplaría en el diagrama, por estar fuera del ámbito de nuestro sistema.
- Puede aparecer en los distintos niveles de DFD para mejorar su comprensión, aunque normalmente sólo aparecerá en el diagrama de contexto.
- Pueden aparecer varias veces en un mismo diagrama, para evitar entrecruzamientos de líneas.
- Suministra información acerca de la conexión del sistema con el mundo exterior.
- Toda entidad externa se identifica con un nombre.



*Figura 3: Entidad externa*

Cuando una entidad externa aparece varias veces en un mismo diagrama, su notación suele ser la siguiente:

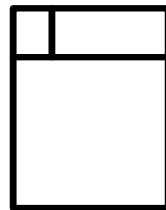


*Figura 4: Entidad externa (2)*

**Proceso:** Conjunto de actividades de negocio que explican qué se hace y cómo se llevan a cabo. Estas actividades transforman o manipulan datos, representan los procedimientos utilizados para transformar los datos. El proceso muestra una parte del sistema que transforma entradas en salidas. Algunos sinónimos comunes para nombrarlo son burbuja, función o transformación. Podemos decir que el proceso se nombra o describe con una sola palabra, frase u oración sencilla. Un buen nombre generalmente consiste en una frase verbo-objeto tal como validar entradas o calcular impuesto. En algunos casos, puede contener el nombre de una persona o un grupo o de una computadora o aparato mecánico, es decir, en ocasiones, el proceso describe quién o qué lo está efectuando, en lugar de describir el proceso mismo; pero este último es un caso muy particular que se aplica para la construcción de los modelos de procesadores.

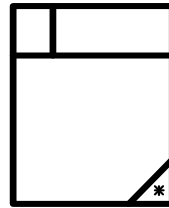
A la hora de identificar un proceso debemos tener en cuenta lo siguiente:

- Cuando un flujo de datos entra en un proceso sufre una transformación. No es origen ni final de datos, sólo es un lugar de transformación de los mismos.
- Un proceso puede transformar un dato en varios.
- Es necesario como intermediario entre una entidad externa y un almacén de datos (veremos a continuación en qué consisten)
- Las entradas y salidas a un proceso son representadas por los flujos de datos.
- Debe ser capaz de generar los flujos de datos de salida a partir de los flujos de datos de entrada y una información local.
- Un proceso se identifica mediante un número y un nombre, que deben ser únicos en el conjunto de los DFDs que representan el sistema.



*Figura 5: Proceso*

**Proceso primitivo:** Es un proceso que no puede ser descompuesto en más subniveles. Suelen tener muy pocos flujos tanto de entrada como de salida y suelen ser sencillos.



*Figura 6: Proceso primitivo*

**Almacén de datos:** Es un lugar físico donde se almacenan datos procesados o desde donde se recuperan para apoyar a un proceso. Un almacén puede corresponder a una tabla en una base de datos, a un archivo en disco, pero también a datos almacenados en otros medios como por ejemplo microfichas, microfilms, o incluso a datos almacenados en papel guardado en un cajón. Los almacenes se conectan a los procesos por medio de los flujos de datos (veremos a continuación en qué consisten).

Los flujos pueden salir o entrar a los almacenes. Un flujo que sale de un almacén se interpreta como una lectura de información del almacén. Un flujo de entrada a un almacén se interpreta como una escritura en el almacén. Esto podría significar que se agregan, se borran o se modifican datos. En cualquiera de los casos es evidente que el almacén cambiará. El proceso que está en el otro extremo del flujo es el encargado de producir el cambio en el almacén.

Una cosa a tener en cuenta es que los almacenes son pasivos, es decir, los datos no viajarán hacia un flujo o desde un flujo a menos que un proceso lo solicite.

Las principales características de los almacenes de datos son las siguientes:

- Representa la información en reposo.
- No puede crear, destruir ni transformar datos.
- No puede estar comunicado directamente con otro almacén o entidad externa.
- No debe estar referido al entorno físico, y por tanto, no se diferencian los ficheros convencionales de las bases de datos.
- Si en un DFD hay un almacén que solo tiene conexión con un único proceso, se dice que es local a dicho proceso, por lo que no debe aparecer en ese nivel, sino en el nivel donde se detalla el proceso.
- Cada almacén debe representarse por primera vez en el nivel más alto en el que sirve de conexión entre dos o más procesos. Además, también se representará en los niveles inferiores.

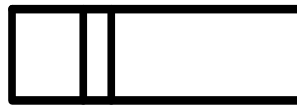


- No se representa la clave de acceso a este almacén, sino la operación que se realiza (lectura, escritura o modificación).



*Figura 7: Almacén de datos*

Cuando una entidad externa aparece varias veces en un mismo diagrama, su notación suele ser la siguiente:



*Figura 8: Almacén de datos (2)*

**Flujo de datos:** Señala el flujo de datos de una entidad externa a un proceso y viceversa, de un proceso a otro, y de un proceso a un almacén de datos y viceversa. Se usa para describir el movimiento de los datos de una parte del sistema a otra. Los flujos tienen una dirección dada por la cabeza de una flecha en cualquier extremo de esta, los flujos que tienen dos cabezas muestran un diálogo, es decir, se representa un pedido y una respuesta en el mismo flujo. Una alternativa a este diálogo es representar dos flujos de datos diferentes, uno que muestre la entrada y otro que muestre la salida. Los flujos de datos en un DFD pueden ser divergentes o convergentes, es decir, que un flujo se puede dividir en varios flujos o varios flujos pueden unirse para formar uno solo.

Cuando el flujo de datos es divergente, puede significar dos cosas: o bien que se estén creando copias de los datos que se están enviando a diferentes partes del sistema, o bien, que el paquete de datos sea tan complejo que se esté dividiendo en paquetes de datos más pequeños, y cada uno de los cuales se esté enviando también a distintas partes del sistema.

Cuando el flujo es convergente quiere decir, que se están uniendo varios paquetes de datos para formar uno más complejo.

Tenemos que tener en cuenta a la hora de identificar un flujo de datos:

- A través de él fluyen datos de estructura conocida.
- Los datos no pueden ser creados ni destruidos por un flujo de datos.
- Sirve para conectar el resto de los componentes de un DFD.
- No es un activador de procesos.

- Cuando un proceso almacena datos, la flecha de flujo de datos se indica en la dirección del almacén de datos y a la inversa si es el proceso el que lee los datos en el almacén.
- Debe tener un nombre que lo identifique.



*Figura 9: Flujo de datos*

Como hemos comentado anteriormente, hay flujos dobles, es decir, que tienen dos cabezas y muestran un diálogo.



*Figura 10: Flujo doble*

Los flujos de datos que conectan componentes deben seguir las siguientes restricciones de conexión:

<i>ORIGEN</i> <i>DESTINO</i>	<i>PROCESO</i>	<i>ALMACÉN</i>	<i>ENTIDAD EXTERNA</i>
<i>PROCESO</i>	SI	SI	SI
<i>ALMACÉN</i>	SI	NO	NO
<i>ENTIDAD EXTERNA</i>	SI	NO	NO

*Tabla 4: Restricciones de los flujos de datos*

## 3. Gestión del proyecto

Definimos gestión de un proyecto como todas las actividades y tareas ejecutadas por una o más personas, con el propósito de planificar y controlar las actividades de otros para alcanzar un objetivo o completar una actividad que no puede ser realizada por otros actuando independientemente.

En esta sección explicaremos las principales áreas de gestión de este proyecto, que son la planificación, la organización y los costes del mismo.

### 3.1. Planificación

La planificación de un proyecto se encarga de clarificar el orden de las tareas y estimar el tiempo necesario para llevarlas a cabo. Se hace dentro de un marco de tiempo limitado al comienzo de un proyecto software, y deberá actualizarse regularmente a medida que progresa el proyecto. Además las estimaciones deberán definir los escenarios del mejor y peor caso, de modo que los resultados del proyecto puedan limitarse.

La planificación se realiza en los siguientes pasos:

- **División del trabajo:** La división del trabajo se lleva a cabo descomponiendo el proyecto en sus objetivos principales, y a su vez dividiendo dichos objetivos en otros de menor importancia y más concretos, hasta identificar el trabajo de desarrollo que se necesita llevar a cabo .
- **Estimación de tiempos:** Sólo es posible comenzar a estimar el tiempo que nos llevará el proyecto cuando hayamos descompuesto adecuadamente cada objetivo en tareas más simples. Lo mejor es centrarse en las tareas de más bajo nivel primero, ya que las predicciones serán mucho más precisas.
- **Identificación de hitos.** Los hitos son objetivos intermedios en el proceso de desarrollo del proyecto y constituyen pasos previos para la consecución de la meta final. Elegirlos correctamente nos ayudará a darnos cuenta de los progresos realizados.
- **Encadenamiento de actividades:** Se da el orden de ejecución de las diferentes tareas que tenemos, teniendo en cuenta si es o no necesario que termine una para que comience la siguiente.

- **Replanificación:** Puede suceder que a medida que avance el proyecto, algunas tareas deban ser replanificadas en tiempo, ya que duren más de lo esperado y se deban tomar decisiones sobre la marcha.

A continuación vemos la planificación hecha mediante un diagrama de Gantt de nuestro software, detallando el tiempo estimado de realización de las diferentes actividades y tareas así como los recursos asignados a cada una de las citadas tareas, y seguidamente el diagrama de Gantt que se cumplió realmente, explicando posteriormente las diferencias.

Vemos el resumen del diagrama:

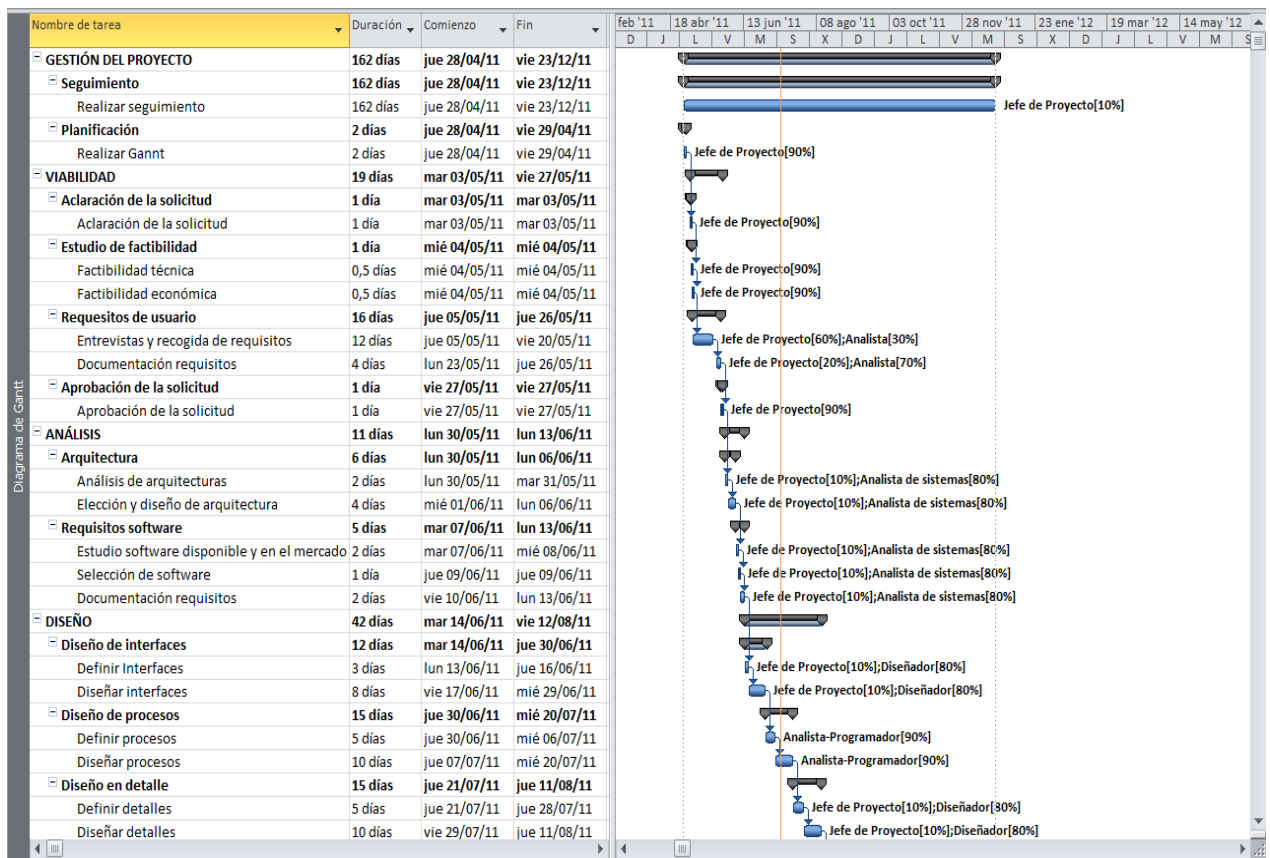


Figura 11: Diagrama Gantt estimado (1)

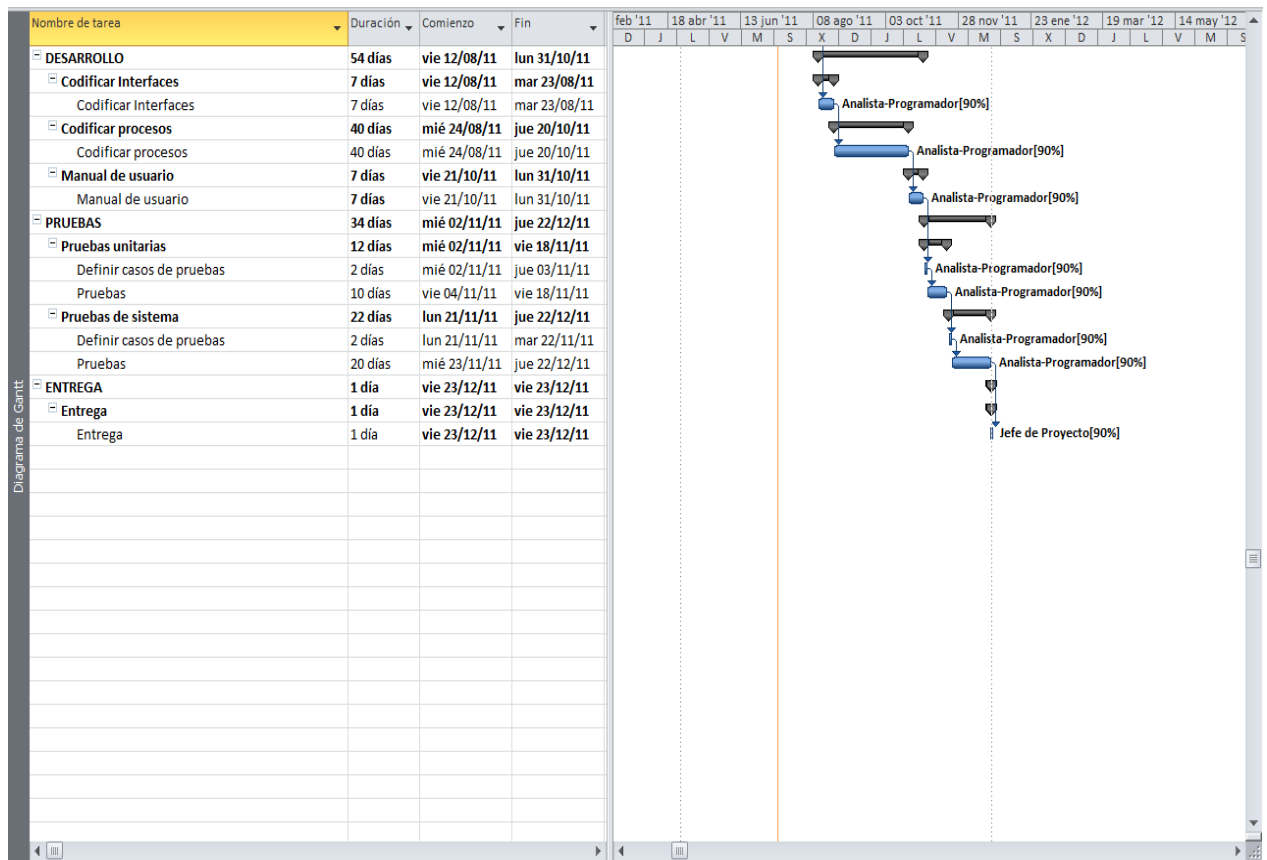


Figura 12: Diagrama Gantt estimado (2)

Vemos ahora el diagrama que se cumplió realmente, resaltando las diferencias con el anterior:

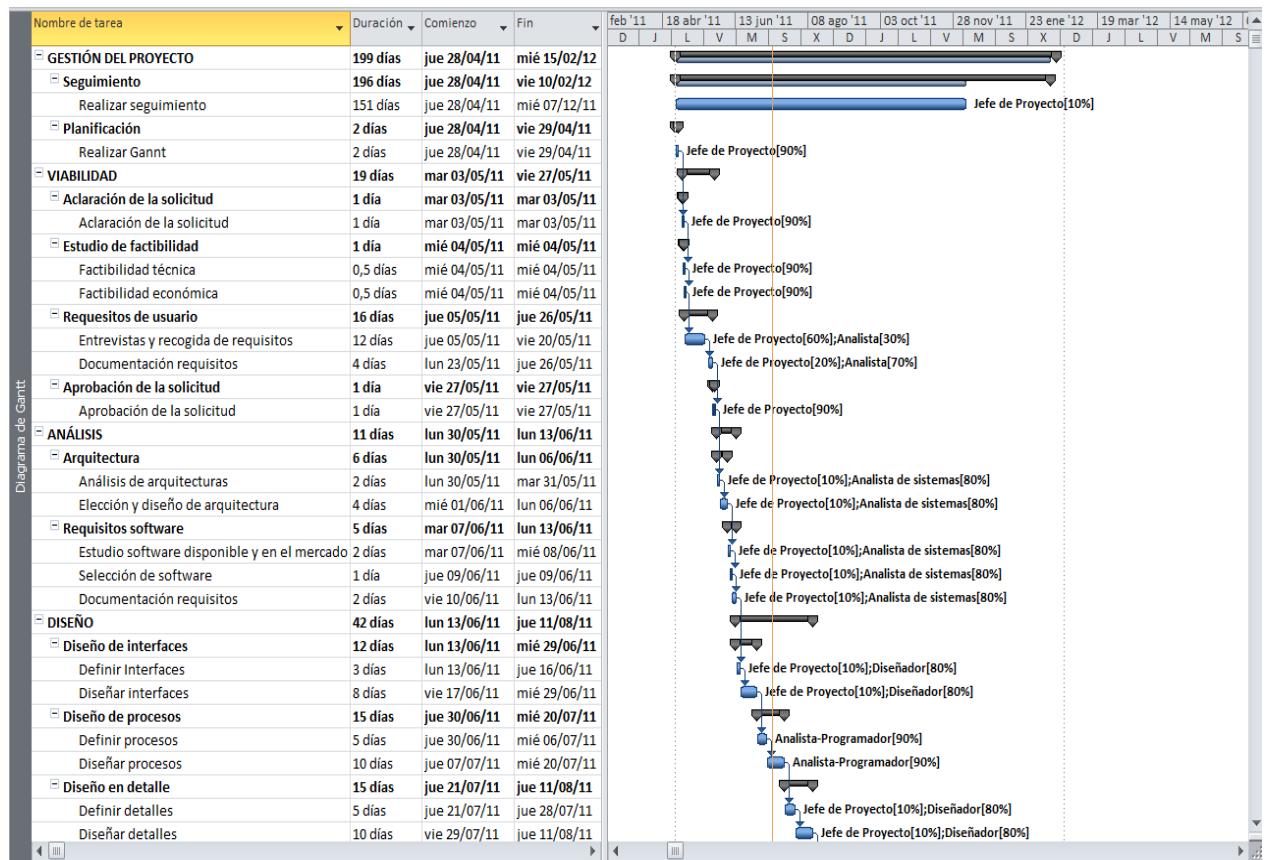


Figura 13: Diagrama Gantt real (1)

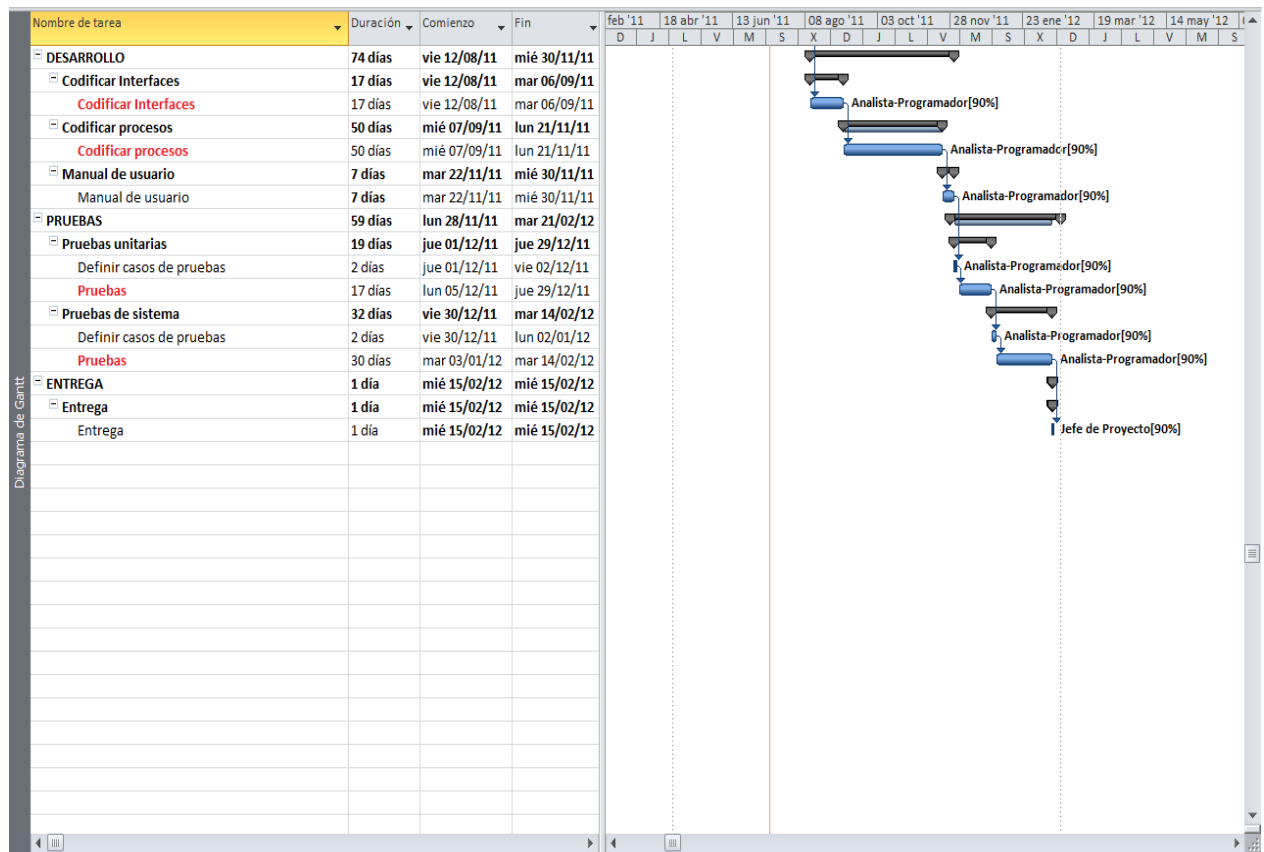


Figura 14: Diagrama Gantt real (2)

### 3.1.1. Comparativa entre la planificación inicial y la planificación final

Como podemos observar, la diferencia de los diagramas radica en las etapas de desarrollo y pruebas. En ambas se ha invertido más tiempo del que inicialmente se había previsto, concretamente 17 días más en cada una, lo que evidentemente nos lleva a un retraso de la entrega del software. La causa de dicho retraso fue la integración de Microsoft Visio, la cual originó más problemas de lo esperado y retrasó los desarrollos. El retraso de esta etapa, origina un retraso en cadena en el comienzo de las siguientes, ya que el modelo utilizado implica que no empezamos una tarea hasta que finaliza la anterior, y a esto hemos de sumarle que también en la etapa de pruebas, se ha necesitado emplear más tiempo del previsto, ya que se han encontrado más partes que corregir de las esperadas, ocasionadas por una gran complejidad en código que no se esperaba y por cumplir en su totalidad los planes de pruebas, el cual no se quiso reducir para compensar el tiempo que ya se llevaba de retraso y así garantizar la calidad del software.

## 3.2. Organización

La organización de un proyecto implica desarrollar una estructura organizacional efectiva y eficiente para asignar y completar las tareas del proyecto, así como establecer las relaciones de autoridad y responsabilidad entre las tareas.

Para este proyecto se ha seguido una organización funcional clásica. Es una jerarquía donde cada empleado tiene un superior claramente establecido y los miembros se agrupan según su especialidad. Vemos el organigrama en la siguiente figura:



*Figura 15: Organigrama*

Sus labores dentro del proyecto son:

- **Jefe de proyecto:** Se encargará principalmente de las etapas de gestión y viabilidad, pero estará presente a modo de seguimiento en todas las tareas planificadas. Es el máximo responsable del proyecto.
- **Analista-Programador:** Estará presente en la toma de requisitos y se encargará del diseño de procesos. También llevará las etapas de desarrollo y pruebas.
- **Analista de sistemas:** Llevará la etapa de análisis, encargándose de la arquitectura y requisitos del software.
- **Diseñador:** Será el encargado de la etapa de diseño, a excepción del diseño de procesos, la cual será responsabilidad del analista-programador.



Se tiene en cuenta que la labor de realización del presente proyecto la ha llevado una única persona física sobre la cual han recaído los diferentes roles que se detallan anteriormente.

### 3.3. Presupuesto del proyecto

Este cálculo se ha realizado en función de las horas dedicadas a cada tarea, los costes indirectos del proyecto, el coste de los equipos informáticos y otros costes directos.

Vemos el desglose a continuación:

- **Costes de personal:**

Categoría	Horas totales de dedicación	Media de dedicación de hombres al mes	Media coste hombre mes	Coste total
Jefe de Proyecto	199,8 horas	18 horas	540,00 €	5.994,00 €
Diseñador	111,68 horas	11,97 horas	239,31 €	2.233,60 €
Analista-Programador	778,5 horas	83,38 horas	347,41 €	19.455,00 €
Analista de Sistemas	52,8 horas	5,66 horas	141,43 €	1.320,00 €
			<b>TOTAL</b>	<b>29.002,60 €</b>

*Tabla 5: Costes de personal*

- El trabajo se compone de un único miembro, el cual desempeña todos los roles del equipo. El horario se fija a razón de seis horas diarias, que hacen un total de treinta horas semanales. Los costes por hora se han calculado según la categoría que desempeñan.

Categoría	Precio por hora
Jefe de Proyecto	30 €
Diseñador	20 €
Analista-Programador	25 €
Analista de Sistemas	25 €

*Tabla 6: Costes por hora según categoría profesional*

- **Costes de equipo:**

Descripción	Coste (Euro)	% Uso dedicado proyecto	Dedicación (meses)	Periodo de depreciación	Coste imputable
<b>Marca:</b> Compaq <b>Ram:</b> 4,00GB <b>Disco duro:</b> 586 GB <b>Procesador:</b> 2.80 GHz	550,00	100	10	60	91,67 €
<b>Total</b>					<b>91,67 €</b>

*Tabla 7: Costes de equipo*

El coste de equipo que se aplica es el coste de amortización del mismo.

- **Otros costes directos:**

Descripción	Empresa	Costes imputable
Parking	Aparca-Seguro	72 €
<b>Total</b>		<b>72 €</b>

*Tabla 8: Otros costes directos*

En otros costes del proyecto, imputamos el precio total que se abonó en concepto de aparcamiento cuando nos desplazamos para la toma de requisitos.

- **Resumen de costes:**

Presupuesto Costes Totales	Presupuesto Costes Totales
Personal	29.003 €
Amortización	92 €
Costes de funcionamiento	72 €
Costes Indirectos (20%)	5.833 €
<b>Total</b>	<b>35.000 €</b>

*Tabla 9: Resumen de costes*

**El presupuesto total de este proyecto asciende a la cantidad de: 35.000 €**

*Tabla 10: Presupuesto total*

## 4. Tecnología utilizada

La herramienta CASE es una aplicación en la cual, para su construcción, como se ha dicho anteriormente, se ha utilizado el lenguaje de programación denominado ‘C #’ o ‘C sharp’.

En esta sección veremos el origen de C#, sus características y la plataforma en la cual se ha llevado a cabo el desarrollo. [6] [7] [10] [11]

### 4.1. Evolución de C

El lenguaje de programación orientado a objetos C parte de un lenguaje anterior, el lenguaje B, escrito por Ken Thompson en 1970, con el objetivo de recodificar el sistema operativo UNIX, que hasta el momento se había programado en ensamblador. A su vez B, fue inspirado en el BCPL de Martin Richard, diseñado tres años antes.

En 1972 es Dennis Ritchie (de los Laboratorios Bell de AT&T) quien diseña finalmente C a partir del B de Thompson, aportando un diseño de tipos y estructuras de datos que consiguen una claridad y eficacia en el lenguaje muy superior.

Paralelamente, en 1980 surge C++ de la mano de Bjarne Stroustrup (también de laboratorios Bell de AT&T). Diseña este lenguaje con el objetivo de añadir a C nuevas características: clases y funciones virtuales (de SIMULA 67), tipos genéricos y expresiones (de ADA), la posibilidad de declarar variables en cualquier punto del programa (de ALGOL 68), y sobre todo, un auténtico motor de objetos con herencia múltiple, que permite combinar la programación imperativa de C con la programación orientada a objetos.

La última variante que ha surgido de C es el moderno C#. En el año 2000, Microsoft presenta su plataforma .NET junto con un nuevo lenguaje, C# (diseñado por Anders Hejlsberg), que servirá de lenguaje principal de la plataforma. C# es un híbrido de C++ y Java que fusiona, principalmente, la capacidad de combinar operadores propia del primero (no incorpora la herencia múltiple) con la plena orientación a objetos del segundo. La orientación a objetos es tal, que el propio programa está encapsulado en una clase. Actualmente C# se encuentra entre los 10 lenguajes más utilizados. A pesar de su corta historia, ha recibido la aprobación del estándar de dos organizaciones: en el 2001 se aprueba el ECMA y en el 2003 el ISO.

## 4.2. Historia de c#

C# (leído en inglés “C Sharp” y en español “C almohadilla”) se diseñó en específico para la plataforma .Net como un lenguaje que permitiera a los programadores migrar con facilidad hacia .Net. Sus principales creadores son Scott Wiltamuth y Anders Hejlsberg, éste último también conocido por haber sido el diseñador del lenguaje Turbo Pascal y la herramienta RAD Delphi. Es un lenguaje de programación orientado a objetos que utiliza clases y estructuras para implementar tipos como formularios Windows Forms, controles de interfaz de usuario y estructuras de datos. También contiene una poderosa biblioteca de clases FCL, que consta de componentes preconstruidos que permiten a los programadores desarrollar aplicaciones con rapidez, además, este lenguaje es apropiado para desarrollar aplicaciones de escritorio, aplicaciones web y aplicaciones móviles, entre otras.

C# es un lenguaje de programación visual controlado por eventos, en el cual se crean programas mediante el uso de un IDE. Con un IDE, un programador puede crear, ejecutar, probar y depurar programas en C#, con lo cual se reduce el tiempo requerido para producir un programa funcional en una fracción del tiempo que llevaría sin utilizar el IDE. La plataforma .Net permite la interoperabilidad de los lenguajes: los componentes de software de distintos lenguajes pueden interactuar como nunca antes se había hecho. Los desarrolladores pueden empaquetar incluso hasta el software antiguo para que trabaje con nuevos programas en C#. Además, las aplicaciones en C# pueden interactuar a través de internet mediante el uso de estándares industriales de comunicación como XML o el SOAP.

## 4.3. Microsoft .NET

La plataforma .NET es la propuesta de Microsoft para el desarrollo de aplicaciones orientadas a objetos, seguras, sencillas de instalar y multiplataforma. No es un lenguaje de programación, es un conjunto de tecnologías software que nos permite el desarrollo de aplicaciones de escritorio, bibliotecas de código, dispositivos móviles, acceso a datos y mucho más, que se ejecutan bajo un entorno de trabajo que se denomina .NET Framework. Es completamente extensible: podemos heredar de sus clases y modificar su comportamiento o implementar sus interfaces para desarrollar clases nuevas.



*Figura 16: Logotipo de Microsoft .net*

El CLR es el núcleo de la plataforma .NET. Es el motor encargado de gestionar la ejecución de las aplicaciones para ella desarrolladas y a las que ofrece numerosos servicios que simplifican su desarrollo y favorecen su fiabilidad y seguridad. Las principales características y servicios que ofrece el CLR son:

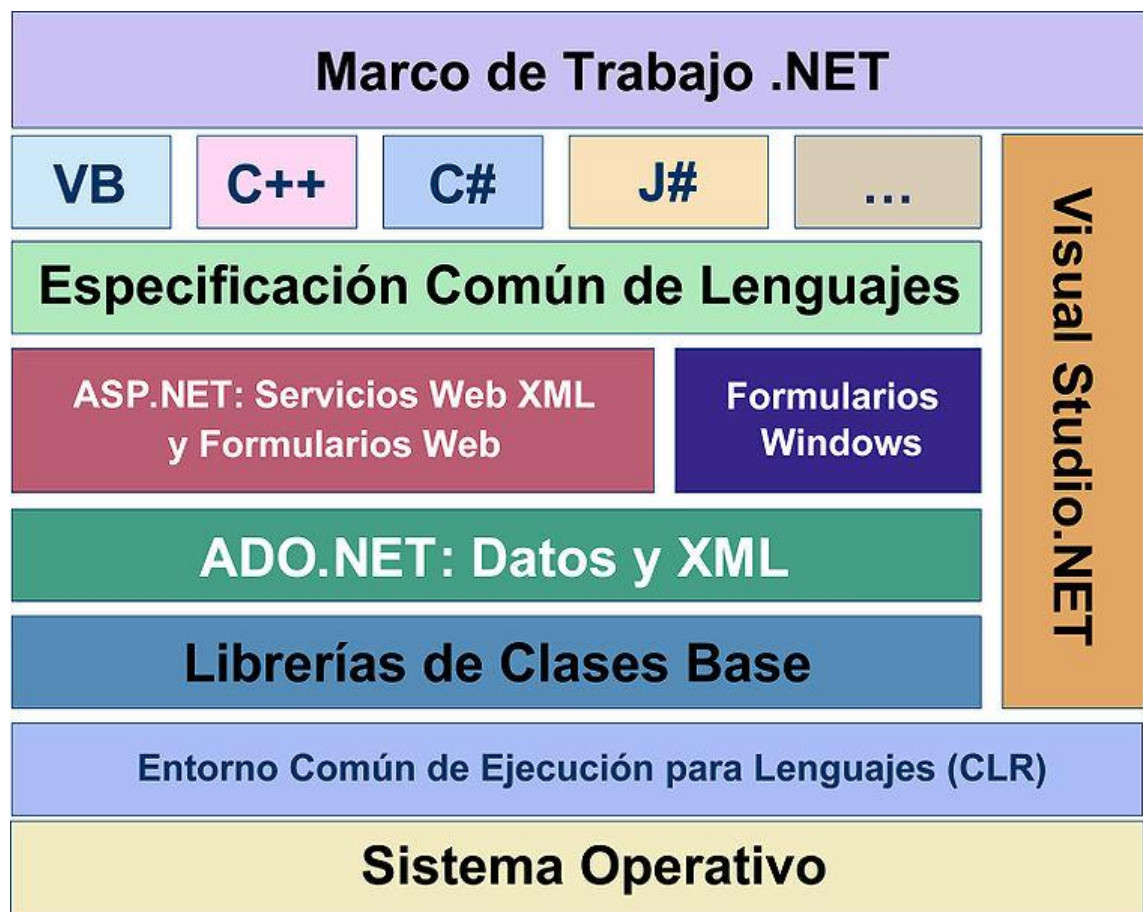
- Modelo de programación consistente: A todos los servicios y facilidades ofrecidos por el CLR se accede de la misma forma: a través de un modelo de programación orientado a objetos. Esto es una diferencia importante respecto al modo de acceso a los servicios ofrecidos por algunos sistemas operativos, en los que a algunos servicios se les accede a través de llamadas a funciones globales definidas en DLLs y a otros a través de objetos (objetos COM en el caso de la familia Windows).
- Modelo de programación sencillo: Con el CLR desaparecen muchos elementos complejos incluidos en los sistemas operativos actuales, como por ejemplo el registro de Windows.
- Eliminación de los problemas con las DLLs: En la plataforma .NET desaparece el problema conocido como “infierno de las DLLs” que se da en los sistemas operativos actuales de la familia Windows, problema que consiste en que al sustituirse versiones viejas de DLLs compartidas por versiones nuevas puede que aplicaciones que fueron diseñadas para ser ejecutadas usando las viejas dejen de funcionar si las nuevas no son compatibles con las anteriores. En la plataforma .NET las versiones nuevas de las DLLs pueden coexistir con las viejas, de modo que las aplicaciones diseñadas para ejecutarse usando las viejas podrán seguir usándolas tras instalación de las nuevas. Esto, obviamente, simplifica mucho la instalación y desinstalación de software.
- Ejecución multiplataforma: El CLR actúa como una máquina virtual, encargándose de ejecutar las aplicaciones diseñadas para la plataforma .NET. Es decir, cualquier plataforma para la que exista una versión del CLR podrá ejecutar cualquier aplicación .NET. Microsoft ha desarrollado versiones del CLR para la mayoría de las versiones de Windows.

- Integración de lenguajes: La integración de lenguajes es tal, que es posible escribir una clase en C# que herede de otra escrita en Visual Basic.NET que, a su vez, herede de otra escrita en C++ por ejemplo.
- Gestión de memoria: El CLR incluye un recolector de basura que evita que el programador tenga que tener en cuenta cuándo ha de destruir los objetos que dejen de serle útiles. Este recolector es una aplicación que se activa cuando se quiere crear algún objeto nuevo y se detecta que no queda memoria libre para hacerlo, caso en que el recolector recorre la memoria dinámica asociada a la aplicación, detecta qué objetos hay en ella que no puedan ser accedidos por el código de la aplicación, y los elimina para limpiar la memoria de “objetos basura” y permitir la creación de otros nuevos. Gracias a este recolector se evitan errores de programación muy comunes como intentos de borrado de objetos ya borrados, agotamiento de memoria por olvido de eliminación de objetos inútiles o solicitud de acceso a miembros de objetos ya destruidos.
- Seguridad de tipos: El CLR facilita la detección de errores de programación difíciles de localizar comprobando que toda conversión de tipos que se realice durante la ejecución de una aplicación .NET se haga de modo que los tipos origen y destino sean compatibles.
- Aislamiento de procesos: El CLR asegura que desde código perteneciente a un determinado proceso no se pueda acceder a código o datos pertenecientes a otro, lo que evita errores de programación muy frecuentes e impide que unos procesos puedan atacar a otros. Esto se consigue gracias al sistema de seguridad de tipos antes comentado, pues evita que se pueda convertir un objeto a un tipo de mayor tamaño que el suyo propio, ya que al tratarlo como un objeto de mayor tamaño podría accederse a espacios en memoria ajenos a él que podrían pertenecer a otro proceso. También se consigue gracias a que no se permite acceder a posiciones arbitrarias de memoria.
- Tratamiento de excepciones: En el CLR todo los errores que se puedan producir durante la ejecución de una aplicación se propagan de igual manera: mediante excepciones.
- Soporte multihilo: El CLR es capaz de trabajar con aplicaciones divididas en múltiples hilos de ejecución que pueden ir evolucionando por separado en paralelo o intercalándose, según el número de procesadores de la máquina sobre la que se ejecuten. Las aplicaciones pueden lanzar nuevos hilos, destruirlos, suspenderlos por un tiempo o hasta que les llegue una notificación, enviarles notificaciones, sincronizarlos, etc.
- Distribución transparente: El CLR ofrece la infraestructura necesaria para crear objetos remotos y acceder a ellos de manera completamente transparente a su localización real, tal y como si se encontrasen en la máquina que los utiliza.
- Seguridad avanzada: El CLR proporciona mecanismos para restringir la ejecución de ciertos códigos o los permisos asignados a los mismos según su procedencia o el usuario que los ejecute. Es decir, puede no darse el mismo nivel de confianza a código



procedente de internet que a código instalado localmente o procedente de una red local; puede no darse los mismos permisos a código procedente de un determinado fabricante que a código de otro; y puede no darse los mismos permisos a un mismo código según el usuario que lo esté ejecutando o según el rol que éste desempeñe. Esto permite asegurar al administrador de un sistema que el código que se esté ejecutando no pueda poner en peligro la integridad de sus archivos, la del registro de Windows, etc.

A continuación vemos un cuadro resumen del entorno de trabajo, con todos sus componentes de la plataforma .NET



*Figura 17: Marco de trabajo de la plataforma .NET*

#### 4.4.1. Compilador

Para el desarrollo y compilación de la aplicación hemos comentado ya que se ha utilizado la herramienta Microsoft Visual Studio 2008.

Microsoft Visual Studio es un IDE para sistemas operativos Windows. Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic .NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros.

Visual Studio permite a los desarrolladores crear aplicaciones, sitios y aplicaciones web, así como servicios web en cualquier entorno que soporte la plataforma .NET (a partir de la versión net 2002). Así se pueden crear aplicaciones que se comuniquen entre estaciones de trabajo, páginas web y dispositivos móviles.

Visual Studio 2008 fue publicado el 17 de Noviembre de 2007 en inglés, mientras que la versión en castellano no fue publicada hasta el 2 de Febrero de 2008.

El nuevo framework (.Net 3.5) está diseñado para aprovechar las ventajas que ofrecen los nuevos sistemas operativos a través de sus subsistemas WCF y WPF. El primero tiene como objetivo la construcción de aplicaciones orientadas a servicios mientras que el último apunta a la creación de interfaces de usuario más dinámicas que las conocidas hasta el momento.

A las mejoras de desempeño, escalabilidad y seguridad con respecto a la versión anterior, se agregan entre otras, las siguientes novedades:

- La mejora en las capacidades de pruebas unitarias permiten ejecutarlas más rápido independientemente de si lo hacen en el entorno IDE o desde la línea de comandos. Se incluye además un nuevo soporte para diagnosticar y optimizar el sistema a través de las herramientas de pruebas de Visual Studio. Con ellas se podrán ejecutar perfiles durante las pruebas para que ejecuten cargas, prueben procedimientos contra un sistema y registren su comportamiento y utilizar herramientas integradas para depurar y optimizar.
- Con VSTO integrado con Visual Studio 2008, es posible desarrollar rápidamente aplicaciones de alta calidad basadas en la interfaz de usuario de Office, que personalicen la experiencia del usuario y mejoren su productividad en el uso de Word, Excel, PowerPoint, Outlook, Visio, InfoPath y Project.
- Visual Studio 2008 permite incorporar características del nuevo WPF sin dificultad tanto en los formularios de Windows existentes como en los nuevos. Ahora es posible actualizar el estilo visual de las aplicaciones al de Windows Vista debido a las mejoras

en MFC y Visual C++. Visual Studio 2008 permite mejorar la interoperabilidad entre código nativo y código manejado por .NET. Esta integración más profunda simplificará el trabajo de diseño y codificación.

- LINQ es un nuevo conjunto de herramientas diseñado para reducir la complejidad del acceso a base de datos. Permite filtrar, enumerar, y crear proyecciones de muchos tipos y colecciones de datos utilizando todos la misma sintaxis, prescindiendo del uso de lenguajes especializados como SQL o XPath.
- Visual Studio 2008 ahora permite la creación de soluciones multiplataforma adaptadas para funcionar con las diferentes versiones de .Net Framework: 2.0. (Incluido con Visual Studio 2005), 3.0 (incluido en Windows Vista) y 3.5 (incluido con Visual Studio 2008).

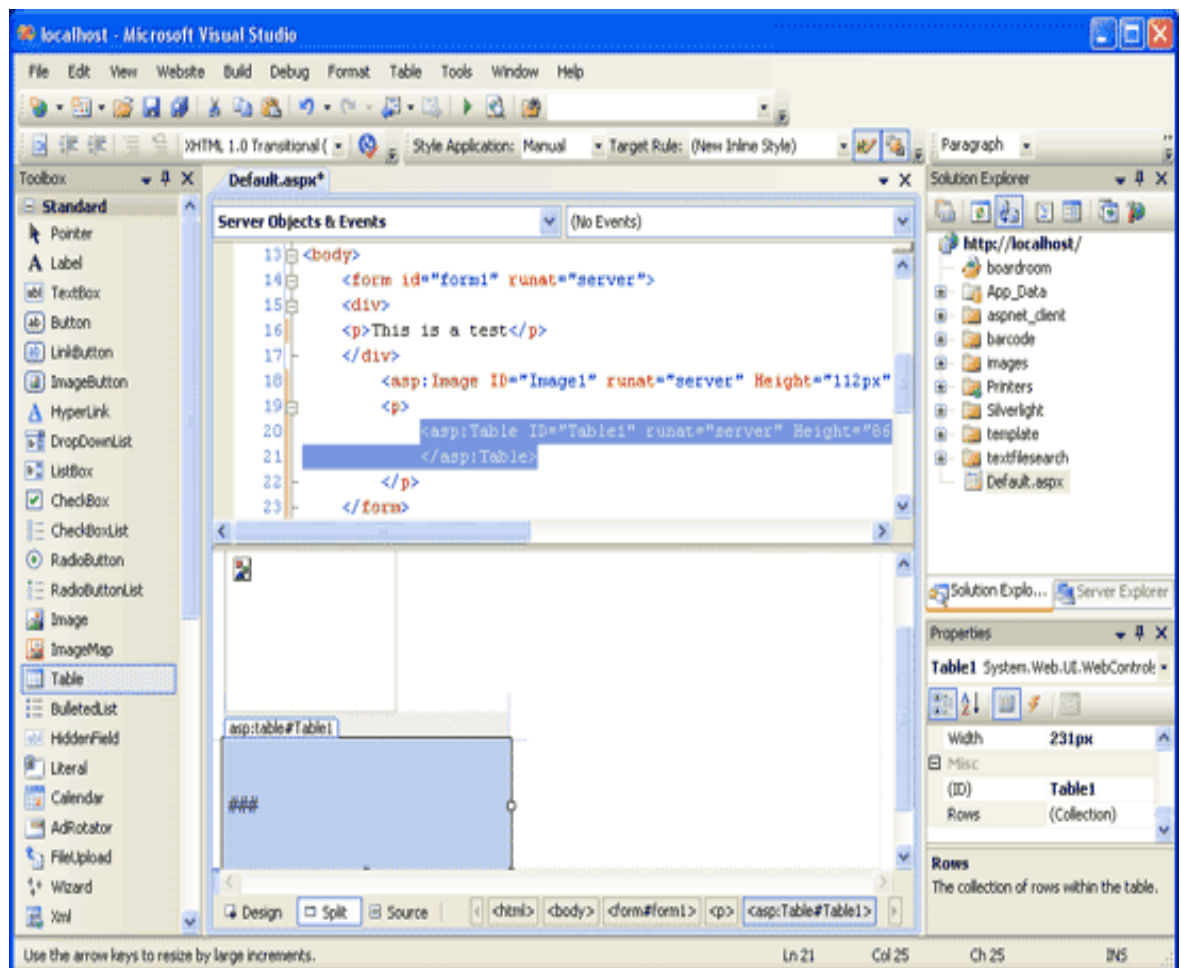


Figura 18: Microsoft Visual Studio2008

## 5. Análisis del sistema

En esta sección se muestra un análisis de nuestro sistema, enumerando los requisitos de usuario pactados con el cliente, los requisitos que un sistema requiere para su instalación y una pequeña guía para realizarla.

### 5.1. Requisitos de usuario

Los requisitos de usuario se definen como aquellas necesidades expresadas por el cliente, las cuales se documentan con el fin de comprobar si lo expresado por éste coincide con el documento de especificación de requisitos de usuario, para así encontrar un canal de comunicación homogéneo sin incertidumbres en la definición de los requisitos.

Para ello, se identificarán los requisitos que ha de satisfacer el nuevo sistema mediante entrevistas, el estudio de los problemas de las unidades afectadas y sus necesidades actuales. Además de identificar los requisitos se deberán establecer las prioridades, lo cual proporciona un punto de referencia para validar el sistema final que compruebe que se ajusta a las necesidades del usuario.

#### 5.1.1. Requisitos funcionales

Los definimos como una característica requerida del sistema que expresa una capacidad de acción del mismo (una funcionalidad).

A continuación mostramos el listado de requisitos.

Identificador	RF1
Nombre	Abrir diagrama
Prioridad	Alta
Descripción	El usuario podrá abrir un diagrama creado con anterioridad. Lo hará desde una caja de diálogo.

*Tabla 11: Requisito funcional 1*

Desde la opción *Abrir*, podremos abrir un diagrama creado previamente en nuestra aplicación. Se nos abrirá una caja de diálogo, en la cual seleccionaremos el documento que queramos, bien sea desde nuestro sistema, desde una unidad de almacenamiento externa o bien

desde alguna unidad de red que tengamos configurada. Únicamente podremos abrir archivos con la extensión \*.vsd

Identificador	RF2
Nombre	Guardar diagrama
Prioridad	Alta
Descripción	El usuario podrá guardar el diagrama activo en la ubicación en la que se encuentra en esos momentos, o bien elegir otra nueva y duplicarlo.

Tabla 12: Requisito funcional 2

Tendremos dos opciones para salvar un diagrama, *Guardar* y *Guardar como*. Con la primera opción, lo que haremos es guardar el diagrama en el mismo lugar donde se encuentra actualmente, sin preguntar, y en caso de que sea la primera vez que lo vayamos a salvar, entonces se ejecutará automáticamente la opción de *Guardar como*, que salvará el documento en la ubicación que nosotros queramos. También podremos directamente utilizar la opción de *Guardar como*, y en caso de que el documento ya esté salvado, entonces lo que haremos será duplicarlo en otra ubicación, o en la misma pero con distinto nombre.

Identificador	RF3
Nombre	Cerrar diagrama
Prioridad	Alta
Descripción	El usuario podrá cerrar el diagrama existente manteniéndose dentro de la aplicación o bien salir por completo de ella. Si se detectaran cambios sin guardar se le preguntará si quiere salvarlos antes.

Tabla 13: Requisito funcional 3

Tendremos dos opciones para cerrar un diagrama, *Cerrar* y *Salir*. Con la primera, lo que haremos es cerrar el diagrama que tenemos abierto actualmente, pero la aplicación seguirá permaneciendo abierta, bien con otros documentos que tuviéramos abiertos o bien sin ninguno. Con la segunda opción, lo que haremos es salir de la aplicación, lo que conlleva cerrar todos los diagramas que tengamos abiertos en ese momento. Para ambas opciones, si hubiera cambios sin guardar en algún diagrama antes de cerrarlo, se dará la opción de salvarlos antes.

Identificador	RF4
Nombre	Imprimir documento
Prioridad	Media
Descripción	Con la opción imprimir, el usuario podrá imprimir el diagrama que tiene abierto, además de poder elegir el número de copias que quiere, si lo quiere en color o no, el tamaño del papel, el tipo, la calidad, la orientación de la página, si lo quiere en color o no... .

Tabla 14: Requisito funcional 4

Con la opción de *Imprimir*, podremos imprimir de todos los documentos que tengamos abiertos, el que tengamos seleccionado en ese momento. Para ello debemos tener configurada una impresora en nuestro sistema, o bien tener acceso a una configurada en red.

Identificador	RF5
Nombre	Nuevo diagrama
Prioridad	Alta
Descripción	El usuario podrá crear un diagrama totalmente nuevo.

Tabla 15: Requisito funcional 5

Desde la opción *Nuevo*, se nos abrirá en la aplicación un diagrama totalmente en blanco. Podremos abrir tantos diagramas nuevos como deseemos. Siempre tendremos el foco en el último que hayamos abierto, los demás quedarán debajo.

Identificador	RF6
Nombre	Cortar objetos
Prioridad	Media
Descripción	El usuario podrá con la opción cortar, mover un objeto u objetos del diagrama y llevarlo al portapapeles de Windows.

Tabla 16: Requisito funcional 6

Con la opción *Cortar*, moveremos lo que tengamos seleccionado en la aplicación al portapapeles, y por lo tanto desaparecerá de la aplicación, dando la sensación de haberlo borrado. Esta opción solo funcionará si previamente tenemos algo seleccionado.

Identificador	RF7
Nombre	Copiar objetos
Prioridad	Media
Descripción	Con la opción copiar, se copiará un objeto u objetos y se llevará al portapapeles de Windows.

*Tabla 17: Requisito funcional 7*

Con la opción *Copiar*, copiaremos lo seleccionado en la aplicación al portapapeles, quedando por lo tanto la aplicación tal y como estaba antes de copiar, dando la sensación de que no ha ocurrido nada. Esta opción sólo funcionará si previamente tenemos algo seleccionado.

Identificador	RF8
Nombre	Pegar objetos
Prioridad	Media
Descripción	Con la opción de pegar, el usuario podrá pegar el contenido el portapapeles en el diagrama.

*Tabla 18: Requisito funcional 8*

Con la opción *Pegar*, se pegará el contenido del portapapeles en el lugar del diagrama que hayamos seleccionado, es el proceso contrario a los dos requisitos anteriores. Esta opción solo funcionará si el portapapeles está relleno, es decir, si antes hemos copiado o cortado algo.

Identificador	RF9
Nombre	Seleccionar todo
Prioridad	Baja
Descripción	Con la opción de seleccionar todo, seleccionaremos todo lo que tengamos en el diagrama.

*Tabla 19: Requisito funcional 9*

Desde la opción *Seleccionar todo*, podremos seleccionar de una sola vez, todo lo que tengamos en el documento activo, bien sean figuras o texto. Una vez seleccionado, tendremos multitud de opciones, como utilizar la opción copiar o cortar, agrandar la selección, o simplemente moverla de posición.

Identificador	RF10
Nombre	Buscar
Prioridad	Baja
Descripción	El usuario podrá buscar cualquier cadena de texto por todos los diagramas del documento que tenga activo en ese momento.

*Tabla 20: Requisito funcional 10*

Con esta opción, podremos buscar cadenas de texto por los diferentes niveles del documento que tengamos activo en ese momento. Si la cadena es encontrada, nos llevará directamente a ese lugar del documento.

Identificador	RF11
Nombre	Eliminar páginas
Prioridad	Media
Descripción	El usuario podrá seleccionar cualquier página del diagrama y eliminarla.

*Tabla 21: Requisito funcional 11*

Con la opción de *Eliminar páginas*, podremos eliminar niveles enteros de un documento que no nos interesen. Seleccionaremos el nivel que queremos eliminar de un listado. Siempre quedará un nivel como mínimo, si sólo se dispone de uno, esta opción no funcionará.

Identificador	RF12
Nombre	Zoom
Prioridad	Media
Descripción	El usuario podrá acercar o alejar la visión del diagrama cuando le sea necesario.

*Tabla 22: Requisito funcional 12*

Con la opción del *Zoom*, podremos ver una panorámica de un solo vistazo del diagrama actual, además de alejar o acercar la visión para poder ver el documento al más mínimo detalle. El cambiar la visión de un diagrama no implica que cambien en los demás niveles de ese mismo documento, cada uno conservará el tamaño que le asignemos individualmente, aunque el cuadro de la panorámica si aparecerá en todos los niveles.

Identificador	RF13
Nombre	Propiedades de las figuras
Prioridad	Media
Descripción	El usuario podrá ver y editar las propiedades que tiene cada figura.

*Tabla 23: Requisito funcional 13*



Desde la opción *Propiedades*, podremos ver y editar las propiedades que tienen las formas de nuestro diagrama. Según la figura se tendrá unas propiedades por defecto u otras. Por ejemplo, de un flujo habrá que definir si es de entrada o salida y además los datos que lleva. Podremos hacerlo sólo de la forma que tengamos seleccionada en ese momento.

Identificador	RF14
Nombre	Ver regla
Prioridad	Baja
Descripción	El usuario podrá mostrar u ocultar la regla en el documento.

*Tabla 24: Requisito funcional 14*

El diagrama aparecerá por defecto con una regla en dos de sus márgenes, una horizontal y otra vertical que nos mostrarán las medidas a escala del dibujo. Por defecto la unidad de medida que aparecerá por defecto serán milímetros. Con esta opción podremos ocultarla o mostrarla cuando nos sea necesario.

Identificador	RF15
Nombre	Texto del diagrama
Prioridad	Baja
Descripción	Con la opción texto, se podrán cambiar las opciones básicas del texto de los diagramas, como el tamaño, el color, tipo de letra, alineación....

*Tabla 25: Requisito funcional 15*

Con esta opción podremos dar formato a las cadenas de texto que se encuentren en el diagrama. Si el texto forma parte de la figura, debemos tener seleccionada la forma para poder cambiar el formato, si no, simplemente debemos seleccionar el texto que queramos. El formato nos permite cambiar opciones básicas del texto.

Identificador	RF16
Nombre	Líneas de las figuras
Prioridad	Baja
Descripción	El usuario podrá modificar la trama de las figuras, así como el grosor, el color, las esquinas, el tamaño, extremos de las líneas, remate, transparencia y además ver una vista previa.

*Tabla 26: Requisito funcional 16*

Desde la opción *Línea*, podremos aplicar distintos formatos a los contornos de las figuras de nuestro diagrama para cambiar su aspecto. Para ello debemos seleccionar antes la forma o

formas a las que queremos aplicar el nuevo formato. Únicamente se verán afectadas las líneas o contornos de las figuras, nunca el interior o relleno.

<b>Identificador</b>	<b>RF17</b>
<b>Nombre</b>	Relleno
<b>Prioridad</b>	Baja
<b>Descripción</b>	El usuario a través de la opción relleno, podrá variar el color, la trama, la transparencia y el estilo tanto del relleno de las figuras como de la sombra.

*Tabla 27: Requisito funcional 17*

Desde la opción *Relleno*, podremos aplicar distintos formatos al interior de las figuras de nuestro diagrama para cambiar su aspecto. También podremos aplicarle sombras. Para ello debemos seleccionar antes la forma o formas a las que queremos aplicar el nuevo formato. Únicamente se verán afectados los rellenos o interiores de las figuras, nunca los contornos. Sólo las formas cerradas pueden tener rellenos y sombras, mientras que las formas abiertas pueden tener relleno, pero no sombras.

<b>Identificador</b>	<b>RF18</b>
<b>Nombre</b>	Exportar a base de datos
<b>Prioridad</b>	Alta
<b>Descripción</b>	La opción exportar a base de datos, dará la opción al usuario de exportar las propiedades que haya definido a cada figura a una base de datos. Podrá hacerlo en distintos formatos.

*Tabla 28: Requisito funcional 18*

Desde la opción *Exportar a base de datos*, podremos mover los datos almacenados en el diagrama hasta una base de datos, de modo que podamos utilizarlos de otras formas. Tendremos que indicar los datos que queremos exportar así como el tipo de base de datos que queremos crear. Sólo se exportarán los datos de las formas que tengamos seleccionadas.

<b>Identificador</b>	<b>RF19</b>
<b>Nombre</b>	Comprobar diagrama
<b>Prioridad</b>	Alta
<b>Descripción</b>	El usuario podrá verificar tanto el diagrama actual detectando errores lógicos como el balanceo entre los distintos niveles.

*Tabla 29: Requisito funcional 19*

Con esta opción podremos ir comprobando a medida que vamos creando el diagrama, si este tiene algún error. Dividiremos ésta comprobación en dos partes. La primera será la opción *Chequear*, con la cual haremos un chequeo únicamente de nivel en el que nos encontramos en ese momento, detectando errores lógicos de construcción, como por ejemplo, que tengamos alguna figura sin nombrar. La segunda opción será la de *Balancear*, que aparte de también detectar algunos errores lógicos, comprobará que el balanceo sea correcto para todos los niveles del diagrama, por tanto, dará igual en qué nivel estemos cuando ejecutemos esta opción. Para asegurarnos que el diagrama está construido de forma correcta, tendremos que ejecutar ambas opciones con éxito. Si se produce algún error en las comprobaciones, se mostrará un listado indicando el tipo de error y el nivel en el que está.

Identificador	RF20
Nombre	Insertar página
Prioridad	Alta
Descripción	El usuario con la opción de insertar página, podrá ir insertando páginas al diagrama e ir configurándolas independientemente.

Tabla 30: Requisito funcional 20

Siempre que creemos un diagrama nuevo, se nos creará únicamente con un nivel. Con la opción *Nueva página*, podremos insertar todos los niveles que deseemos en el diagrama y su configuración será siempre independiente.

Identificador	RF21
Nombre	Insertar figuras
Prioridad	Alta
Descripción	El usuario mediante una paleta de figuras situada al lado del documento, podrá ir arrastrándolas para ir dibujando el diagrama.

Tabla 31: Requisito funcional 21

Podremos ir insertando figuras en nuestro diagrama mediante una paleta que por defecto nos aparece en la parte izquierda de la aplicación. Para ello debemos seleccionar una forma con el ratón y arrastrarla hasta dentro del diagrama. Esto se hará siempre de una en una. Otra forma de insertar figuras será utilizando la opción de *Copiar*, aunque para ellos debemos haber insertado antes otra figura. Con esta segunda opción, sí podremos insertar al diagrama varias formas a la vez.

Identificador	RF22
Nombre	Agregar datos a formas
Prioridad	Alta
Descripción	El usuario podrá ir introduciendo datos sobre las figuras de los diagramas, que luego podrá exportar a una base de datos. Primero agregará los campos de datos a las formas y, a continuación, los valores.

Tabla 32: Requisito funcional 22

Podremos ir agregando datos a las formas del diagrama. Por defecto cada tipo de forma tendrá predefinidos unos tipos de datos, que nosotros podremos ir rellenando, modificando, eliminando o agregando nuevos. Estos datos son los que luego podremos exportar a las bases de datos. Para ello dispondremos de la opción *Propiedades*, o bien seleccionado una forma, haciendo clic con el botón derecho sobre ella y entrando en Datos → Datos de formas. Dado que no todas las formas tienen los mismos tipos de datos definidos, para evitar errores debemos hacer esto con solo una forma cada vez.

Identificador	RF23
Nombre	Ventanas activas
Prioridad	Media
Descripción	El usuario podrá ver todas las ventanas activas dentro de la aplicación, e ir pasando de unas a otras.

Tabla 33: Requisito funcional 23

Cada vez que abramos un diagrama nuevo o uno existente, se nos abrirá una ventana nueva en nuestra aplicación. Mediante la opción *Ventana*, podremos ver todas las ventanas que tenemos abiertas en ese momento, ya que se irán poniendo unas encima de otras y no podremos saberlo de un solo vistazo, a no ser que la minimicemos. Si pulsamos sobre una de ellas, se nos superpondrá a las demás, y la tendremos como la ventana activa en ese momento.

Identificador	RF24
Nombre	Informes
Prioridad	Media
Descripción	El usuario podrá ver dos tipos de informes. El primero con todos los flujos que hay en el diagrama y un segundo con todas las formas.

Tabla 34: Requisito funcional 24

Podremos generar dos tipos de informes dentro de nuestra aplicación. El primero corresponderá a la opción *Flujos Actuales*, en el cual se nos abrirá una ventana emergente mostrándonos todos los flujos que tenemos en nuestro diagrama, indicándonos en qué nivel se encuentran. El segundo informe corresponderá a la opción *Informe*, en el que se nos generará un

informe con todas las figuras del diagrama, bien de un sólo nivel, o bien de todos al completo y podremos guardarlo y visualizarlo en distintos formatos.

<b>Identificador</b>	<b>RF25</b>
<b>Nombre</b>	Avance y retroceso
<b>Prioridad</b>	Media
<b>Descripción</b>	El usuario podrá moverse por los distintos niveles del diagrama.

*Tabla 35: Requisito funcional 25*

Podremos ir pasando de un nivel a otro del diagrama mediante unas flechas de avance y retroceso que tendremos en la barra de herramientas. Estas flechas también se encontrarán en la parte inferior del diagrama, donde también tendremos otras para ir directamente al primer o último nivel y así no tener que pasar por todos obligatoriamente.

<b>Identificador</b>	<b>RF26</b>
<b>Nombre</b>	Paleta de formas
<b>Prioridad</b>	Media
<b>Descripción</b>	El usuario podrá ocultar o mostrar la paleta de formas cuando desee.

*Tabla 36: Requisito funcional 26*

Por defecto, en la parte izquierda de la aplicación, tendremos una paleta con todas las figuras que necesitamos para construir un diagrama. Esta paleta se puede ocultar mediante el aspa que tiene en la parte superior derecha. Para volver a mostrarla, tendremos la opción *Paleta*, que volverá a hacer que aparezca de nuevo.

<b>Identificador</b>	<b>RF27</b>
<b>Nombre</b>	Eliminar formas
<b>Prioridad</b>	Alta
<b>Descripción</b>	El usuario podrá eliminar las formas del diagrama

*Tabla 37: Requisito funcional 27*

Podremos eliminar las figuras que no necesitemos del diagrama cuando queramos. Tendremos que seleccionar la figura o las figuras a eliminar, y pulsar el botón *Supr* de nuestro teclado. Si queremos eliminar todas las figuras a la vez, podremos utilizar la opción *Seleccionar todo* y a continuación pulsar el botón de *Supr*.

<b>Identificador</b>	<b>RF28</b>
<b>Nombre</b>	Explosionar procesos
<b>Prioridad</b>	Alta
<b>Descripción</b>	El usuario podrá explosionar los procesos del diagrama

*Tabla 38: Requisito funcional 28*

Podremos explosionar los procesos del diagrama que necesitemos para una mejor comprensión de lo que queremos representar. Esto lo haremos con el ratón, mediante el botón secundario entraremos en el menú Formato→ Comportamiento, y en la pestaña doble clic, marcaremos la opción *Ir a la página*, donde pondremos el nivel en el cual queremos hacer la explosión. Siempre tienen que ser procesos, nunca procesos primitivos. Mediante la opción balancear, podremos comprobar si lo hemos hecho correctamente.

<b>Identificador</b>	<b>RF29</b>
<b>Nombre</b>	Configurar página
<b>Prioridad</b>	Media
<b>Descripción</b>	El usuario podrá configurar las características de la página

*Tabla 33: Requisito funcional 29*

Desde la opción *Configurar página*, podremos configurar propiedades de cada página del diagrama, como la impresión, tamaño, formatos.... Cada vez que insertemos una página nueva en el diagrama, se nos abrirá esta opción automáticamente. La configuración de cada página o nivel del diagrama es independiente de las demás, pero siempre se aplicará a la página que tengamos activa en ese momento.

<b>Identificador</b>	<b>RF30</b>
<b>Nombre</b>	Sombra de las formas
<b>Prioridad</b>	Baja
<b>Descripción</b>	El usuario podrá dar formato de sombra a las figuras

*Tabla 39: Requisito funcional 30*

Desde la opción *Sombra*, podremos aplicar formato de sombra a las figuras de nuestro diagrama. Para ello debemos seleccionar antes la forma o formas a las que queremos aplicar el nuevo formato.

### 5.1.2. Requisitos no funcionales

Los definimos como una característica requerida del sistema, del proceso de desarrollo, del servicio prestado o de cualquier otro aspecto del desarrollo, que señala una restricción del mismo.

Estos requisitos tratan de acotar el entorno de rendimiento, seguridad, protección de datos y de calidad bajo el cual el producto debe ser plenamente operativo.

Identificador	RNF1
Nombre	Rendimiento
Prioridad	Alta
Descripción	La aplicación deberá respetar un umbral razonable de estabilidad y compatibilidad.

*Tabla 40: Requisito no funcional 1*

La herramienta deberá ser razonablemente estable y compatible con las plataformas, entornos y protocolos más comúnmente utilizados tanto en el ámbito académico como en el profesional.

Identificador	RNF2
Nombre	Protección de la información
Prioridad	Alta
Descripción	El contenido de un proyecto debe ser tratado siempre de manera confidencial.

*Tabla 41: Requisito no funcional 2*

Tanto el conocimiento que un equipo de proyecto deposita en uno de sus proyectos, como la información que hay en él, no deben nunca ser filtrada sin autorización a terceros.

Identificador	RNF3
Nombre	Diseño de interfaz
Prioridad	Media
Descripción	La interfaz debe facilitar el uso de la herramienta al usuario.

*Tabla 42: Requisito no funcional 3*

El diseño de la interfaz deberá minimizar todo lo posible los movimientos necesarios del usuario para efectuar una acción, y su organización debe reunir de la manera más lógica todas las funcionalidades requeridas en la situación actual (ocultando o inhabilitando el resto de ellas) en el menor número posible de ventanas y menús que puedan hacer farragosa la navegación.

## 5.2. Requisitos del sistema

Los requisitos mínimos que ha de tener nuestro sistema para un correcto funcionamiento son los siguientes:

- Se recomienda que el equipo tenga 128 megabytes (MB) de RAM o más.
- 1,5 gigabytes (GB) de espacio de disco duro.
- En cuanto al sistema operativo:
  - Microsoft Windows 2000 Service Pack 3 (SP3) o posterior.
  - Microsoft Windows Server 2003 Service Pack 1 (SP1) o posterior.
  - Microsoft Windows XP Service Pack 2 (SP2) o posterior.
  - Windows Vista.
  - Windows 7.
- Tener instalado Microsoft .NET Framework 2.0 o superior
- Tener instalado Microsoft Visio 2003 ó 2007.

Es muy importante tener en cuenta el tipo de sistema operativo que tenemos a la hora de instalar el software enumerado anteriormente, es decir, si se ejecuta sobre una versión de 32 o de 64 bits. Es muy posible que si realizamos la instalación con distintas versiones, la aplicación no funcione de manera correcta.

## 5.3. Instalación

Los pasos para instalar *CASE-FREE 2008* en nuestro sistema son los siguientes:

En primer lugar debemos ejecutar el archivo *setup*, aceptando lo que nos vaya preguntando. Esto nos instalará automáticamente la aplicación en nuestro sistema.

El segundo paso será hacer una copia del archivo MV3 (plantilla de formas) a la carpeta de plantillas de Microsoft Visio, que generalmente se encuentra en la siguiente ruta:

C:\Archivos de programa\Microsoft Office\Office12\3082.

Con estos dos sencillos pasos queda instalada la aplicación en nuestro sistema. Puede que la instalación se haga correctamente sin cumplir los requisitos del sistema o sin hacer la copia de la plantilla de formas, pero la aplicación no funcionará correctamente.

Para desinstalarlo, lo haremos desde panel de control, en quitar o eliminar programas.



## 6. Ejemplo utilizando CASE-FREE 2008

En este apartado se ha desarrollado un ejemplo práctico para comprobar de qué manera la herramienta da solución a un problema. Primero lo definimos y luego vemos la solución dada. [ACE+03].

### 6.1. Enunciado

Una empresa aeronáutica desea mecanizar la gestión de mantenimiento de aviones de acuerdo a lo siguiente:

1. El cliente realiza una petición de presupuesto de reparación de un avión, señalando las deficiencias observadas.
2. El cliente recibe un presupuesto que se realiza teniendo en cuenta los costes de los componentes. Dicho presupuesto se guarda.
3. Si el cliente entrega el documento de aceptación del presupuesto, éste es aceptado. Se actualiza el archivo de presupuestos y se hace llegar el avión a los hangares de la compañía emitiéndose un albarán de depósito y se comienza la reparación. Al equipo de reparación se le pasa una orden de trabajo.
4. Una vez terminado el trabajo, el equipo de reparación pasa una nota de trabajo. Inmediatamente después de recibir la nota, se activa la emisión de una factura que es entregada al cliente.
5. Un equipo de presupuestos se encarga de actualizar periódicamente los componentes.

### 6.2. DFD de Contexto

El DFD de contexto de este sistema tiene tres entidades externas, la entidad externa *Cliente*, que se identifica en los cuatro primeros puntos y además tiene cinco flujo de datos con el proceso *0 Gestionar Mantenimiento*. El primero es *Petición Presupuesto*, identificado en el punto 1. El segundo es *Presupuesto Inicial*, identificado en el punto 2. En el punto 3 identificamos dos flujos, *Aceptación Presupuesto* y *Albarán Depósito*. Por último, en el punto 4, identificamos el flujo *Factura*.

La entidad externa *Cliente*, se representa con dos iconos para mejorar la legibilidad del diagrama.

La siguiente entidad externa es *Equipo Reparación*, que tiene dos flujos, *Orden de Trabajo* y *Nota de Trabajo*.

La tercera entidad es *Equipo Presupuestos*, que proporciona información sobre el precio de los componentes mediante el flujo *Precio Componente*.

Vemos el diagrama completo a continuación:

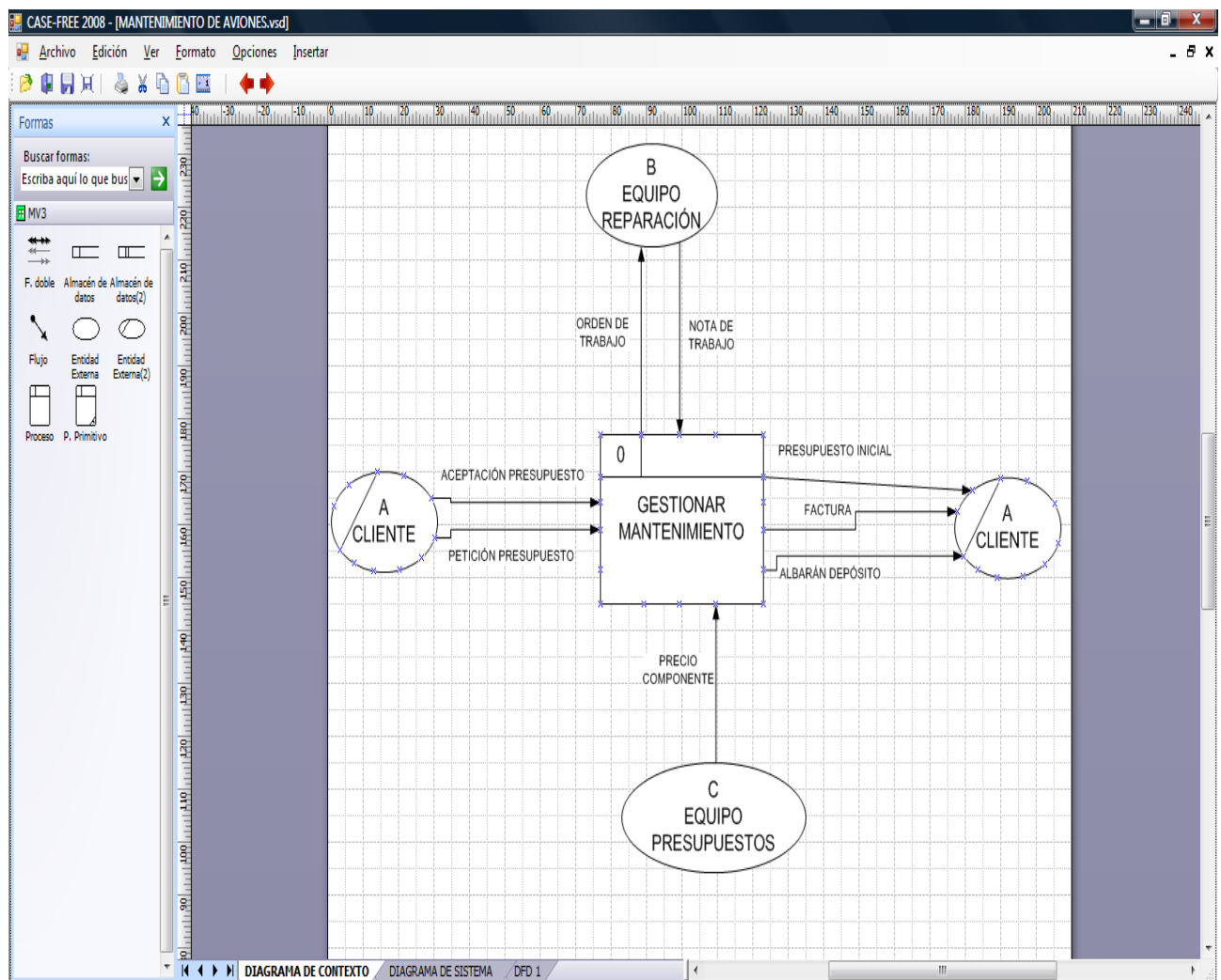


Figura 19: Diagrama de contexto

## 6.3. DFD de Sistema

El DFD de sistema presenta tres procesos. *Gestionar Presupuesto*, que incluye todo lo relacionado con la generación, emisión y aceptación de un presupuesto y que será explotado para tratarlo con más detalle en un nuevo diagrama; *Actualización Componentes*, cuya funcionalidad es actualizar periódicamente el precio de los componentes, y *Emitir Factura*, que se encarga de emitir las facturas.

Identificamos dos almacenes, el primero *Componentes*, que guarda la información referente a los componentes y sus precios, será utilizado por el proceso *Gestionar Presupuesto* para elaborar un presupuesto, y será actualizado por *Actualización Componentes*, y el segundo *Presupuestos*, que guarda los presupuestos pedidos y aceptados para que el proceso *Emitir Factura* genere las facturas.

Vemos el diagrama completo a continuación:

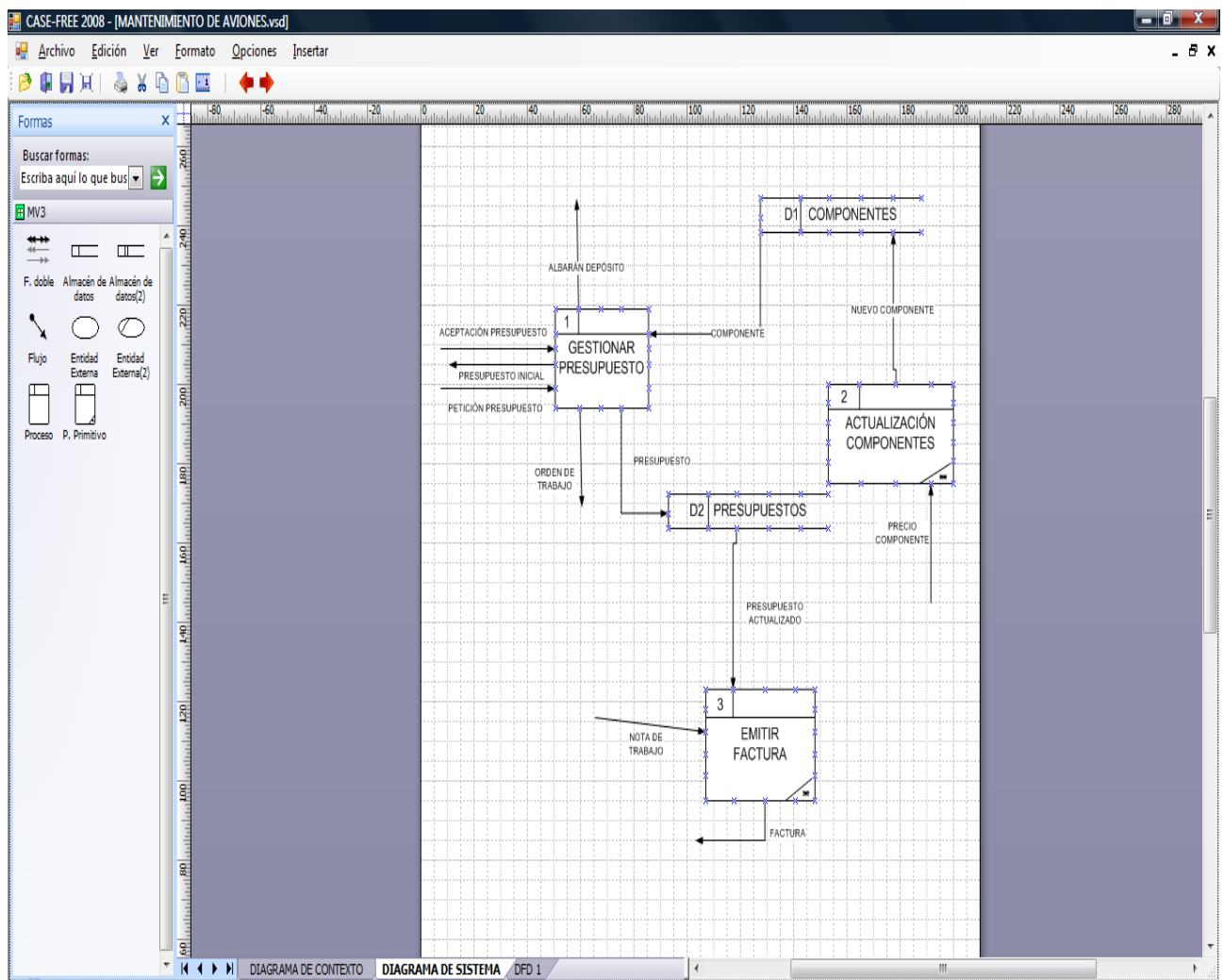


Figura 20: Diagrama de sistema

## 6.4. DFD 1

En este diagrama se trata en detalle el proceso *Gestionar Presupuesto* del DFD de sistema. Se identifican dos procesos, uno es *Tratar Presupuestos*, que automatiza todo el proceso generado hasta que se acepta el presupuesto, recibe la petición de presupuesto y genera y guarda el presupuesto inicial; el segundo es *Tratar Presupuestos*, que automatiza la gestión de un presupuesto entregado, recibe la aceptación del presupuesto como flujo de entrada y dispone de tres flujos de salida, el presupuesto actualizado, el albarán y la orden de trabajo.

Vemos el diagrama completo a continuación:

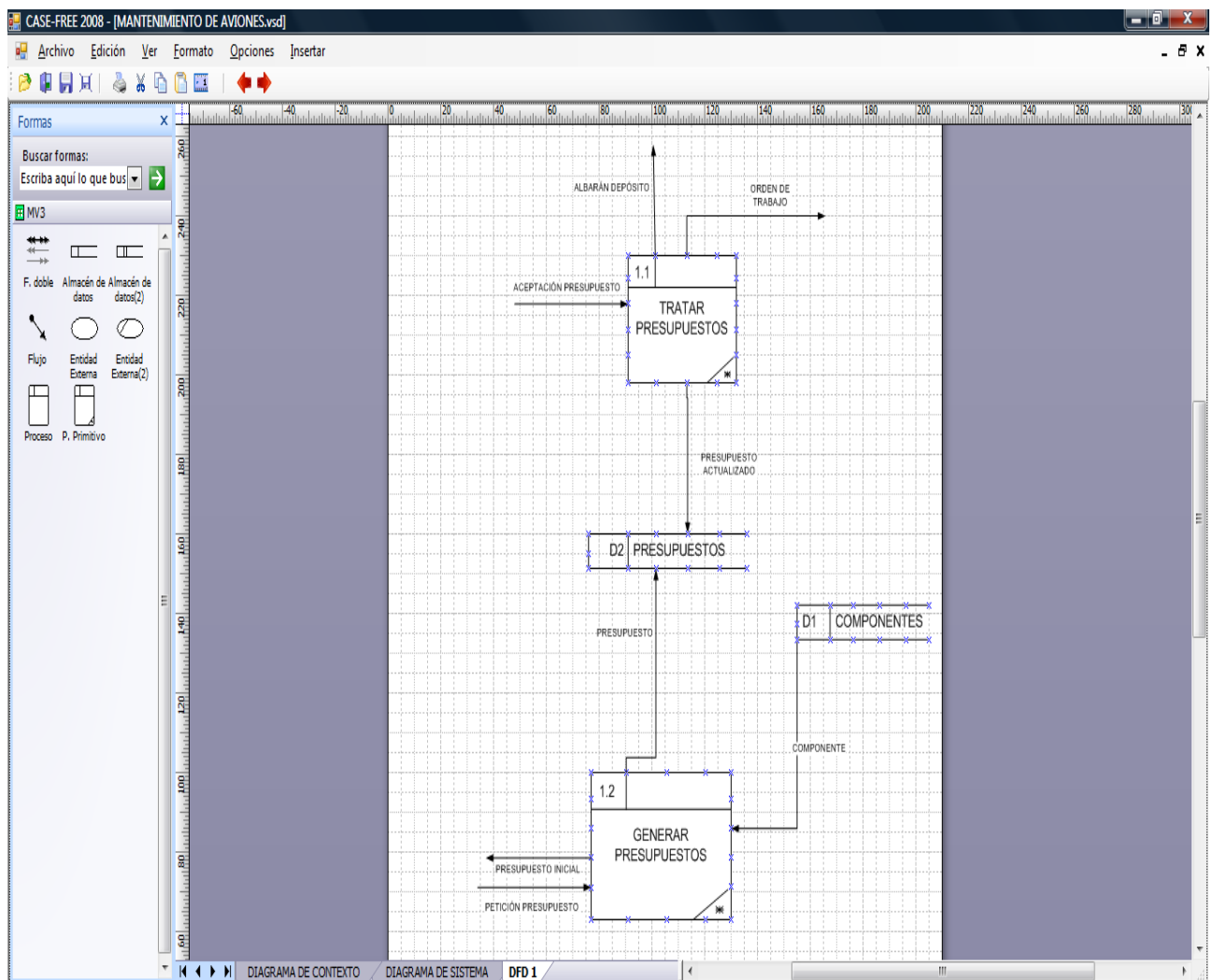


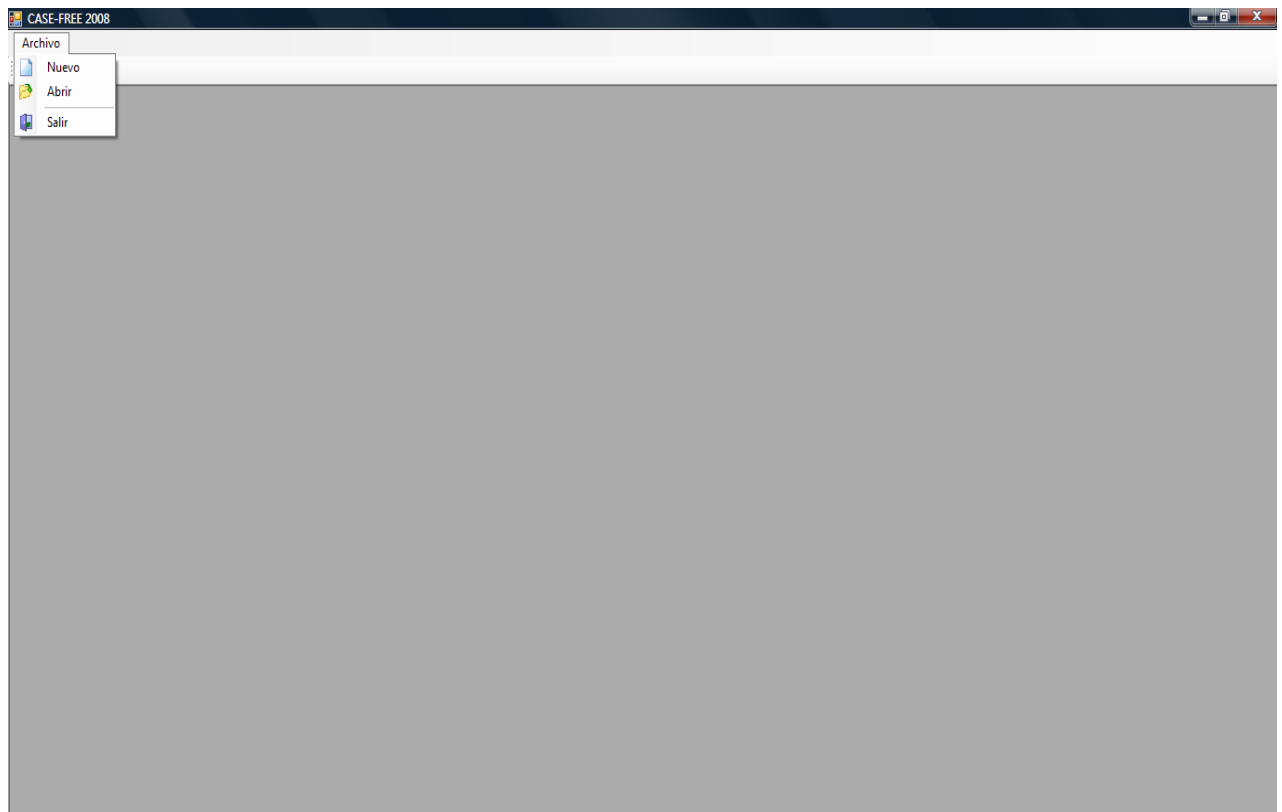
Figura 21: DFD 1

## 7. Manual de usuario

En este punto se explica el funcionamiento de la aplicación, las opciones que nos ofrece y para qué sirven cada una de ellas.

### 7.1. Iniciar sesión

Cuando entramos por primera vez en la aplicación, podemos observar que sólo disponemos de tres acciones dentro del menú Archivo, que son *Nuevo*, *Abrir* y *Salir*. Con cualquiera de las dos primeras entraremos en la aplicación y ya podremos ver el resto de opciones disponibles, con la opción *Salir*, se cerrará el editor.



*Figura 22: Vista nada más entrar en la aplicación*

## 7.2. Conectar figuras entre sí

Para conectar dos figuras del diagrama entre sí, siempre vamos a utilizar como figura de conexión el flujo, bien el simple o bien el doble. Para asegurarnos de que las figuras están conectadas correctamente, arrastraremos un extremo del flujo hacia el centro de la forma a la que lo queremos conectar, y un contorno rojo debe rodear a esta forma, si no vemos el contorno rojo las formas no estarán bien conectadas. Otra forma de conectar dos figuras puede ser directamente a los puntos de conexión que aparecen cuando acercamos el flujo a la forma a la que lo queremos conectar, si aparece un contorno rojo sobre estos puntos, las formas están bien conectadas. Debemos indicar en el primer valor del flujo si es de entrada o salida (E/S) con respecto al proceso en el caso del flujo simple.

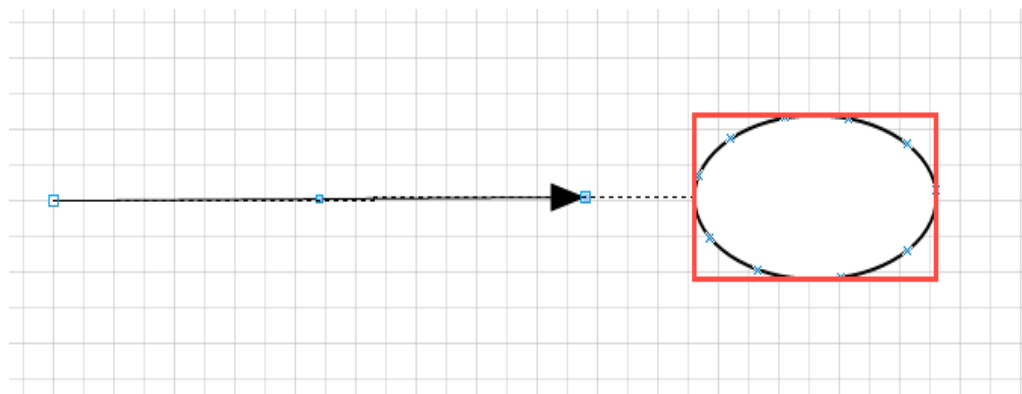


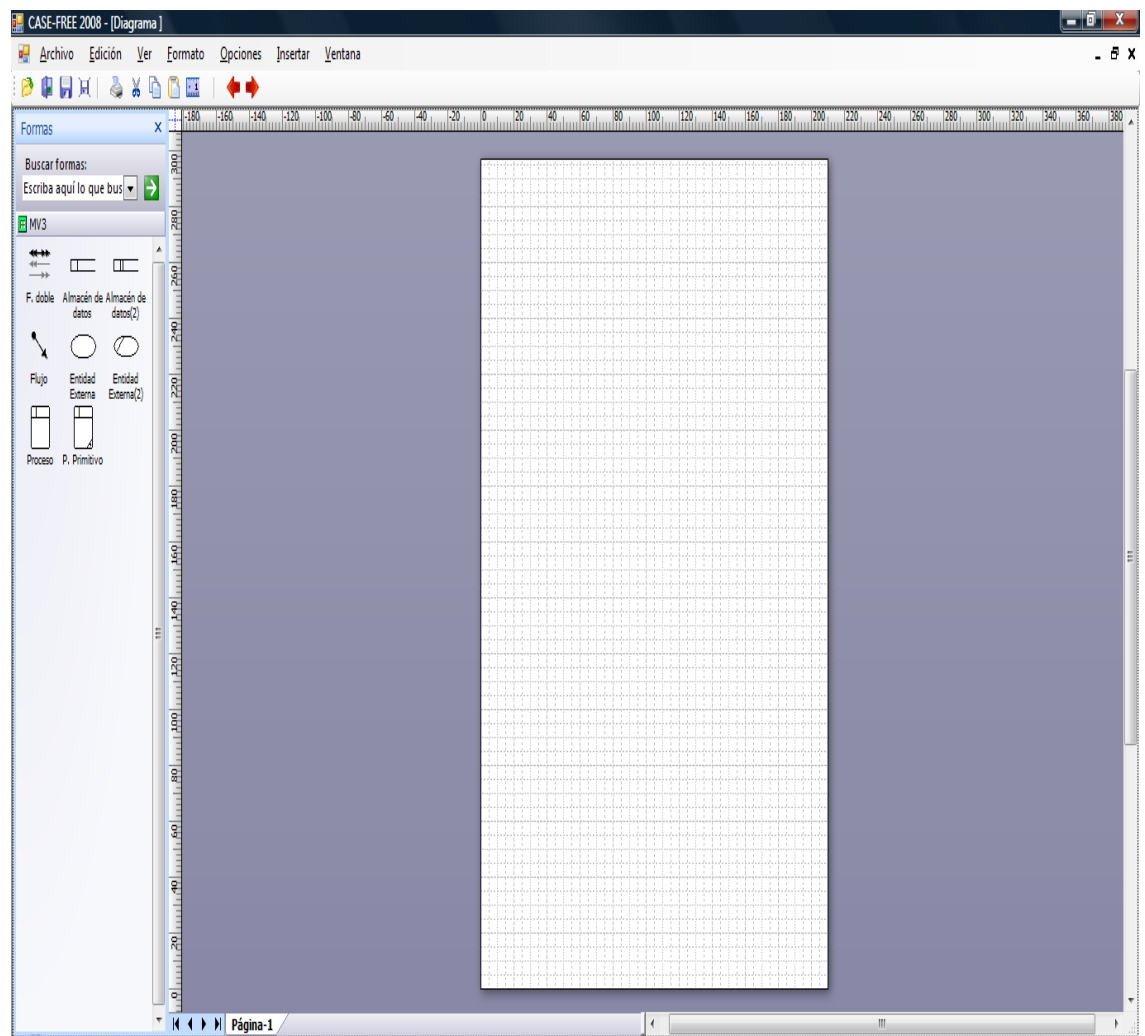
Figura 23: Conexión de dos formas

## 7.3. Opciones de menú

Iremos viendo ahora las opciones del menú una a una.

### ➤ Archivo

- **Nuevo:** Esta opción abre un nuevo diagrama en blanco. Se podrán abrir todos los diagramas nuevos que se quieran. Cada vez que abramos uno, se irá superponiendo la los demás y siempre quedará activo el último abierto. Como nombre por defecto aparecerá siempre *Diagrama*.



*Figura 24: Vista nuevo diagrama*

- **Abrir:** Esta opción abre una caja de diálogo para recuperar un diagrama creado anteriormente. Se mostrarán únicamente las carpetas y los archivos con extensión .vsd, ya que la aplicación siempre los guardará con esa misma. Podremos seleccionar el diagrama tanto del disco duro, como de unidades externas de almacenamiento o unidades de red a las que tengamos acceso.



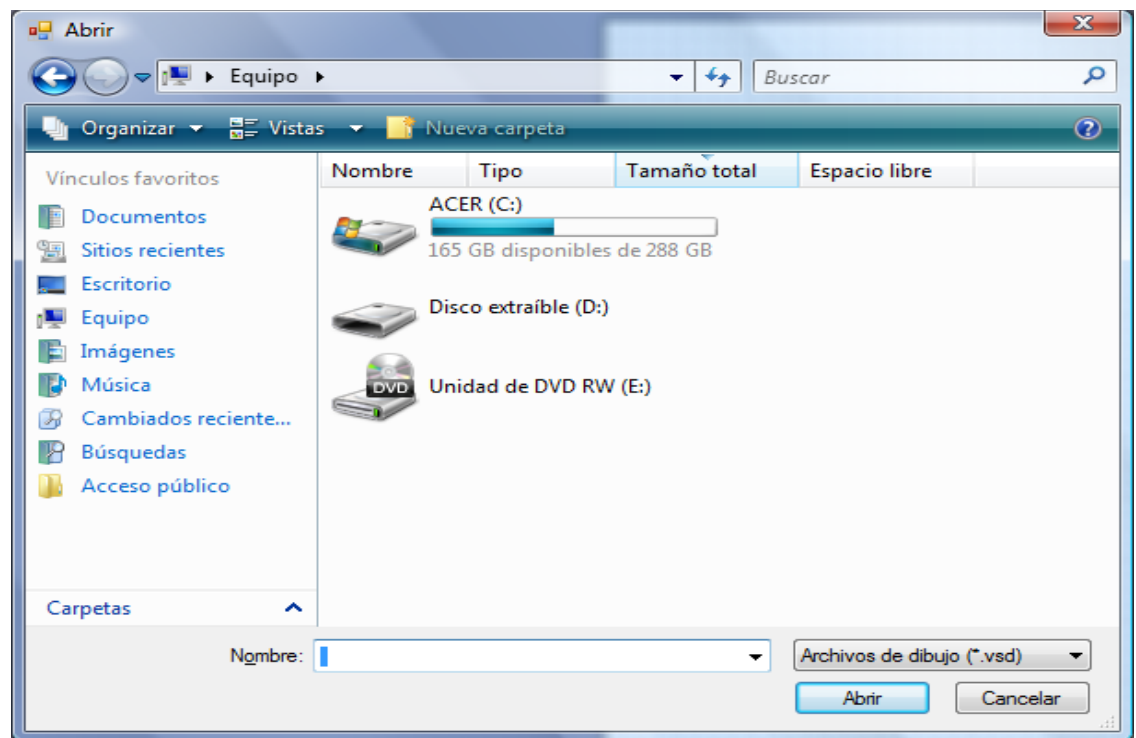


Figura 25: Vista de la opción abrir

- **Configurar página:** Esta opción nos permite configurar únicamente la página que tenemos activa. Se nos abrirá un cuadro de diálogo con seis pestañas, cuyas opciones son las siguientes:
  - **Configurar impresión:**
    - ❖ **Tamaño del papel:** Enumera los tamaños de papel que admite la impresora en uso, así como los tamaños estándar de la industria.
    - ❖ **Orientación del papel:** Permite especificar la orientación del papel como vertical u horizontal.
    - ❖ **Configurar:** Abre el cuadro de diálogo de *Configurar impresión* con las siguientes opciones:
      - **Tamaño:** Enumera los tamaños de papel que admite la impresora en uso.
      - **Origen:** Enumera los orígenes del papel específicos de la impresora de destino, como bandejas de papel, cintas, alimentación manual y otros.
      - **Orientación:** Especifica la orientación del papel como Vertical u Horizontal.
      - **Izquierda:** Especifica la distancia desde el borde izquierdo del papel hasta el margen izquierdo.
      - **Superior:** Especifica la distancia desde el borde superior del papel hasta el margen superior.

- **Derecha:** Especifica la distancia desde el borde derecho del papel hasta el margen derecho.
- **Inferior:** Especifica la distancia desde el borde inferior del papel hasta el margen inferior.
- **Centrar horizontalmente:** Centra un dibujo que es más pequeño que el ancho del papel entre los márgenes izquierdo y derecho.
- **Centrar verticalmente:** Centra un dibujo que es más pequeño que el alto del papel entre los márgenes superior e inferior o bordes del papel.
- **Ajustar a:** Reduce o aumenta el contenido del dibujo en un porcentaje especificado del tamaño real solamente para la impresión; el tamaño real de la página de dibujo no cambia.
- **Ajustar a (hojas de ancho por hojas de alto):** Ajusta cada dibujo en el trabajo de impresión al número de hojas especificado a lo ancho y a lo largo. Según el número de páginas y márgenes que especifique, el dibujo se aumenta o se reduce. Si las páginas a lo ancho y a lo largo no son proporcionales a las dimensiones del dibujo, solamente se utilizan las hojas necesarias para mantener las proporciones del dibujo.
- ❖ **Cuadrícula:** Se imprime la cuadrícula que se ve en la ventana de dibujo.
- ❖ **Vista previa:** Muestra una miniatura con los valores establecidos actualmente en el papel de la impresora y en la página de dibujo para que trabajen conjuntamente.
- ❖ **Papel de la impresora:** Muestra la configuración actual del papel de la impresora, incluidos el tamaño y la orientación.
- ❖ **Página de dibujo:** Muestra la configuración de la página de dibujo, incluidos el tamaño y la orientación.
- ❖ **Zoom para imprimir:** Muestra la reducción o ampliación que haya especificado.
- **Tamaño de página:**
  - ❖ **El mismo tamaño que el papel de la impresora:** Hace coincidir el tamaño de la página de dibujo con el del papel de la impresora, especificado en la ficha Configurar impresión.
  - ❖ **Tamaño predefinido:** Establece el tamaño de la página de dibujo a un tipo de papel y tamaño predefinidos. Al seleccionar otro tipo de papel predefinido (estándar, métrico, etc.), la lista se actualiza para mostrar los tamaños disponibles. Si la orientación de página es vertical, primero se muestra el ancho y después el alto, si es horizontal, se muestra el alto seguido del ancho.
    - **Estándar:** Incluye tamaños de papel estándar en unidades de EE.UU
    - **Métrico (ISO):** Incluye tamaños de papel estándar en unidades métricas.
    - **Ingeniería ANSI:** Incluye tamaños de papel estándar de ingeniería ANSI en unidades de EE.UU.

- **De arquitectura ANSI:** Incluye tamaños de papel estándar de arquitectura ANSI en unidades de EE.UU.
- ❖ **Tamaño personalizado:** Configura la página de dibujo con un tamaño personalizado.
- ❖ **Ajustar al contenido del dibujo:** Configura la página de dibujo para que tenga el tamaño suficiente para contener las formas de la página actual.
- ❖ **Orientación de página:** Especifica la orientación de la página de dibujo como vertical u horizontal.
- ❖ **Vista previa:** Muestra una miniatura con los valores establecidos actualmente en el papel de la impresora y en la página de dibujo para que trabajen conjuntamente.
- ❖ **Papel de la impresora:** Muestra la configuración actual de papel de la impresora tal como está establecida en la ficha Configurar impresión.
- ❖ **Página de dibujo:** Muestra la configuración actual de la página de dibujo tal como está establecida en la ficha Tamaño de página.
- ❖ **Zoom para imprimir:** Muestra la reducción o el aumento tal como está establecido en la ficha Configurar impresión.
- **Escala de dibujo:**
  - ❖ **Sin escala (1:1):** Haremos clic en esta opción para crear un dibujo.
  - ❖ **Escala predefinida:** Haremos clic en esta opción para ajustar la página de dibujo a una escala predefinida.
    - **De arquitectura:** Incluye medidas en unidades de EE.UU.
    - **Ingeniería civil:** Incluye medidas en unidades de EE.UU.
    - **Métrica:** Incluye escalas métricas estándar.
    - **Ingeniería mecánica:** Incluye escalas estándar de ingeniería.
  - ❖ **Escala personalizada:** Haremos clic en esta opción para establecer una relación personalizada del tamaño de la página de dibujo con dimensiones físicas.
  - ❖ **Tamaño de página (en unidades de medida):** Establece el tamaño del mundo real de lo que esté dibujando.
  - ❖ **Vista previa:** Muestra una miniatura con los valores establecidos actualmente en el papel de la impresora y en la página de dibujo para que trabajen conjuntamente.
  - ❖ **Papel de la impresora:** Muestra la configuración actual de papel de la impresora tal como está establecida en la ficha Configurar impresión.
  - ❖ **Página de dibujo:** Muestra la configuración actual de la página de dibujo tal como está establecida en la ficha Tamaño de página.
  - ❖ **Zoom para imprimir:** Muestra la reducción o el aumento tal como está establecido en la ficha Configurar impresión.

- **Propiedades de página:**
  - ❖ **Tipo:** Enumera las propiedades para la página de dibujo o para la página de fondo.
  - ❖ **Nombre:** Para cambiar el nombre de la página de dibujo, escribiremos un nombre que contenga un máximo de 31 caracteres.
  - ❖ **Fondo:** Enumera los nombres de las páginas de fondo del dibujo.
  - ❖ **Unidades de medida:** Elegiremos la unidad de medida que deseemos aplicar a las regletas de la página.
- **Diseño y enrutamiento:**
  - ❖ **Estilo:** Establece el estilo predeterminado de enrutamiento de los de la página.
  - ❖ **Dirección:** En los estilos de enrutamiento de flujos Diagrama de flujo, Árbol, Organigrama o Sencillo podemos establecer la dirección o flujo del diagrama.
  - ❖ **Separar:** Especifica si las líneas de flujos superpuestas se deben separar.
    - **Líneas no relacionadas:** Especifica que si las líneas de los flujos se superponen y los flujos no se conectan a formas comunes, se deben separar.
    - **Todas las líneas:** Especifica que si las líneas de un flujo se superponen, se deben separar.
    - **Ninguna línea:** Especifica que las líneas de flujo no se deben separar.
    - **Con estilo de enrutamiento:** El estilo de enrutamiento de un flujo concreto especifica cómo se separan las líneas superpuestas.
  - ❖ **Superposición:** Especifica si las líneas de flujo que están separadas se deben superponer.
    - **Líneas relacionadas:** Especifica que si las líneas de flujo no se superponen y los flujos están conectados a formas comunes, se deben superponer.
    - **Todas las líneas:** Especifica que si las líneas de flujo están próximas unas de otras, se deben superponer.
    - **Ninguna línea:** Especifica que las líneas de flujo no se deben superponer.
    - **Con estilo de enrutamiento:** El estilo de enrutamiento de un conector especifica cómo se superponen las líneas separadas.
  - ❖ **Apariencia:** Especifica si los nuevos flujos que se agreguen al dibujo deben ser rectos o curvados.
    - **Rectos:** Especifica que los nuevos flujos deben ser rectos.
    - **Curvados:** Especifica que los nuevos flujos deben ser curvados.
  - ❖ **Agregar saltos de línea a:** Especifica cómo se agregan los saltos de línea a los flujos de la página.
    - **Ninguno:** Especifica que la página no contiene saltos de línea.
    - **Líneas horizontales:** Especifica que los saltos de línea sólo aparecen en las líneas de flujo horizontales.

- **Líneas verticales:** Especifica que los saltos de línea sólo aparecen en las líneas de flujo verticales.
- **Última línea enrutada:** Especifica que los saltos de línea aparecen en el flujo que se ha agregado o ajustado más recientemente.
- **Última línea mostrada:** Especifica que los saltos de línea aparecen en el flujo situado en la parte superior del orden de apilamiento.
- **Primera línea mostrada:** Especifica que los saltos de línea aparecen en el flujo situado en la parte inferior del orden de apilamiento.
- ❖ **Estilo de salto de línea:** Especifica la apariencia que tienen los saltos de línea de esta página.
- ❖ **Tamaño vertical:** Especifica el alto de los saltos de línea de los segmentos de línea verticales.
- ❖ **Tamaño horizontal:** Especifica el ancho de los saltos de línea de los segmentos de línea horizontales.
- ❖ **Mover las otras formas al colocar:** Especifica que las formas de la página se mueven automáticamente para que se pueda colocar una forma que se arrastra a la página o que se mueve o cambia de tamaño en ésta.
- ❖ **Habilitar división de conectores:** Esta opción está relacionada con los siguientes valores de las formas de la ficha Comportamiento del cuadro de diálogo Comportamiento: Un conector puede dividirse por formas y Una forma puede dividir conectores.
- ❖ **Espaciado:** Abre el cuadro de diálogo Espaciado del diseño y el enrutamiento en el que podemos especificar el espacio entre formas, el tamaño promedio de éstas, el espacio entre conectores, así como el espacio entre flujos y formas.
  - **Espacio entre formas:** Especifica la cantidad de espacio entre formas colocables.
  - **Tamaño promedio de las formas:** Especifica el tamaño promedio de las formas del dibujo.
  - **Conector a conector:** Especifica la cantidad mínima de espacio entre flujos.
  - **Conector a forma:** Especifica la cantidad mínima de espacio entre flujos y formas.
- **Sombras:**
  - ❖ **Estilo:** Establece el estilo de sombra para la página.
  - ❖ **Vista previa:** Muestra una vista previa de las Sombras seleccionadas.
  - ❖ **Tamaño y posición:** Especifica la posición de las sombras con respecto a las formas. Aumento especifica el tamaño de la sombra en relación con la forma.
  - ❖ **Dirección:** Especifica el ángulo de una sombra oblicua.

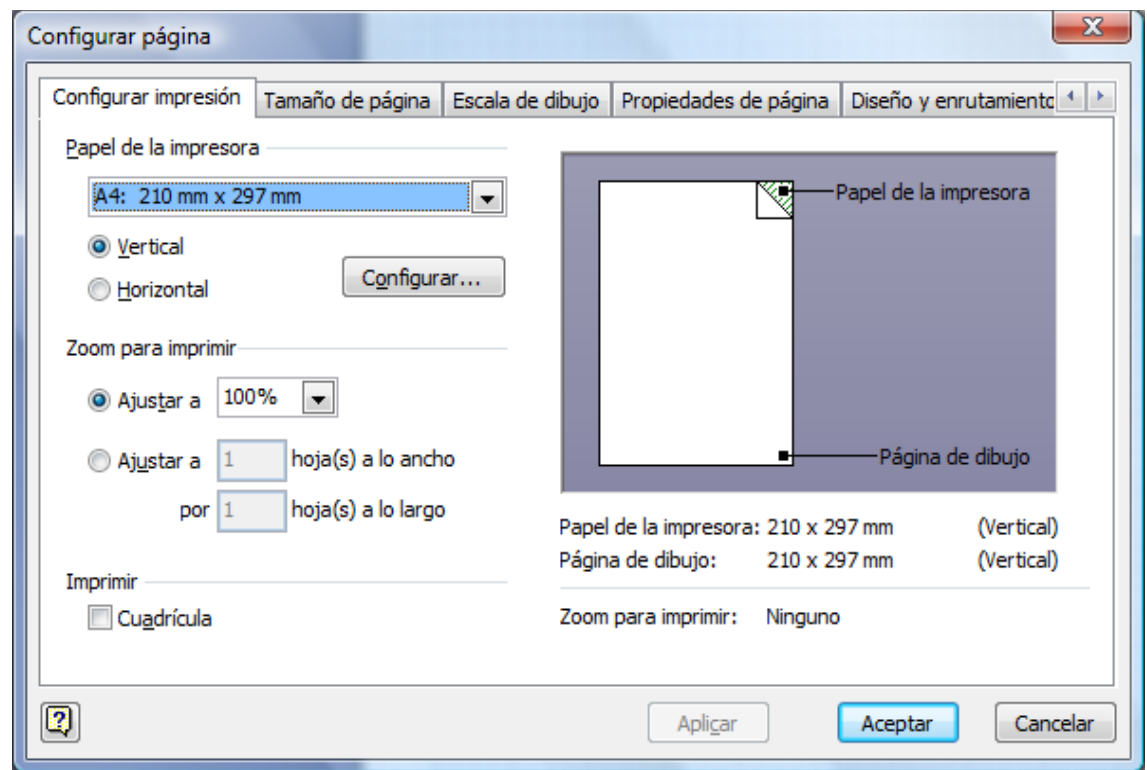


Figura 26: Vista de la opción Configurar página

- **Guardar:** Esta opción se utiliza para salvar el diagrama que tengamos activo en ese momento. Si el diagrama no fue salvado nunca, entonces llamaremos a la opción Guardar como, que está explicada a continuación. Por el contrario, si el diagrama ya ha sido guardado anteriormente y pulsamos sobre esta opción, se guardará con el mismo nombre y en la misma ubicación que ya tenía. Se podrá guardar en una carpeta de la unidad de disco duro, en una ubicación de red, en un disco, en el escritorio o en otra ubicación de almacenamiento. El proceso siempre será el mismo elija la ubicación que elija. Al ejecutar esta acción, puede dar la sensación de que no ha ocurrido nada, pero el diagrama se actualizó con los últimos cambios introducidos.
- **Guardar Como:** Esta opción se utiliza para salvar el diagrama activo. Se abrirá una caja de dialogo para darle un nombre y seleccionar la ubicación donde queramos guardarlo. Se podrá guardar en una carpeta de la unidad de disco duro, en una ubicación de red, en un disco, en el escritorio o en otra ubicación de almacenamiento. El proceso siempre será el mismo elija la ubicación que elija. Si va a guardar un diagrama que no se ha guardado anteriormente, el cuadro de diálogo *Guardar como*, aparecerá automáticamente al hacer clic en la opción *Guardar*. Es muy útil para hacer copias de un diagrama ya existente.

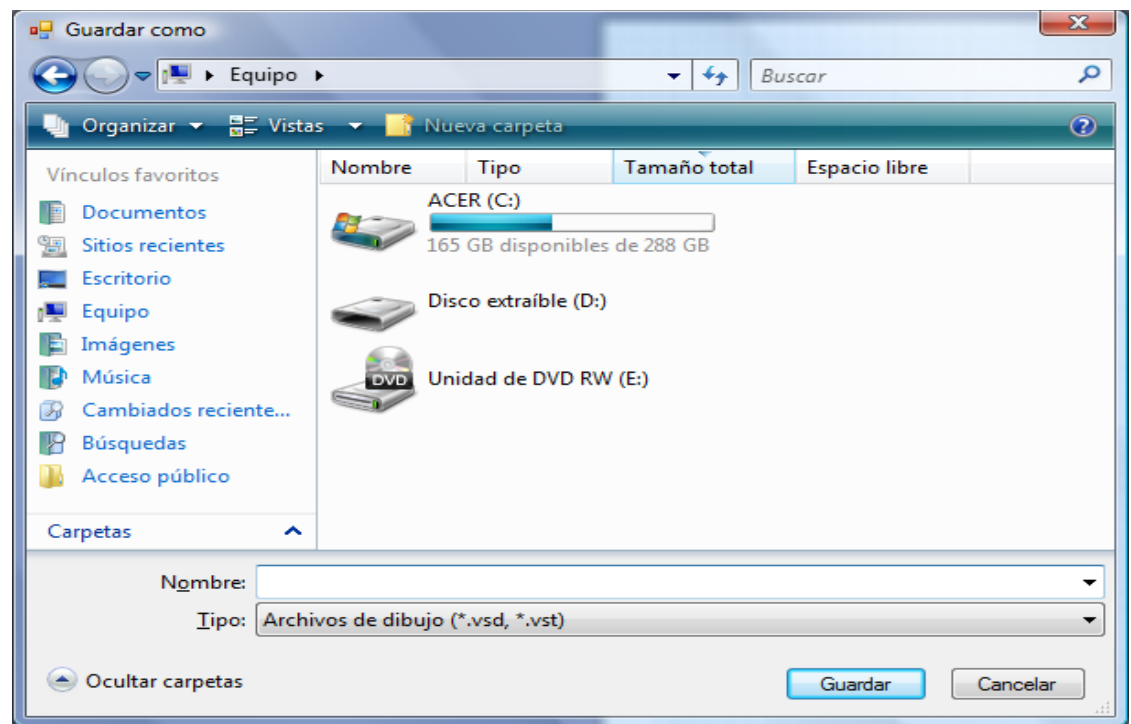


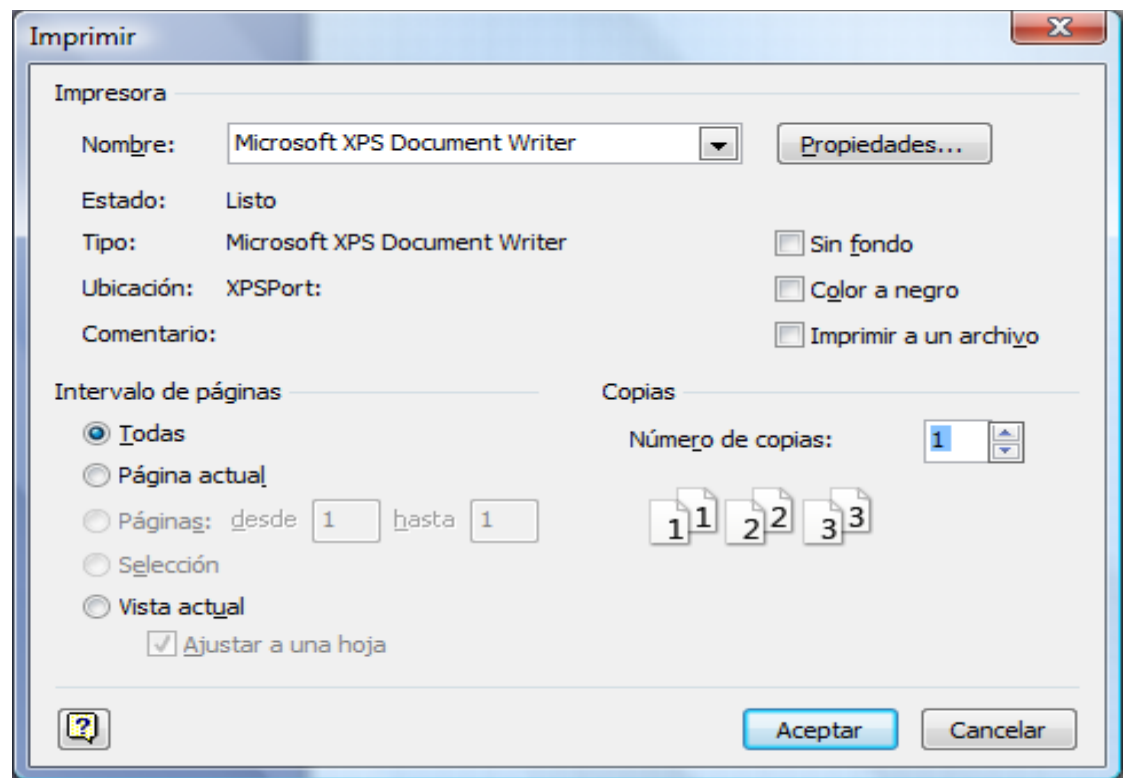
Figura 27: Vista de la opción Guardar como

- **Imprimir:** Esta opción imprime el diagrama activo. Se abrirá una caja de diálogo en la cual podremos ver o configurar las siguientes opciones:
  - **Nombre:** Muestra el nombre de la impresora actual. Para cambiarla, debemos seleccionar otra de la lista.
  - **Propiedades:** Abre un cuadro de diálogo en el que podemos especificar propiedades de la impresora, como el tamaño del papel, la orientación y la resolución de impresión. Las opciones disponibles dependen de la impresora. Debemos tener en cuenta que estas opciones se aplicarán únicamente al trabajo de impresión en curso.
  - **Estado:** Determina si la impresora está disponible y si se trata de la impresora predeterminada.
  - **Tipo:** Especifica el tipo de impresora en el que se va a imprimir.
  - **Ubicación:** Especifica la ubicación de la impresora (su puerto o su ubicación de red).
  - **Comentario:** Especifica información adicional acerca de la impresora.
  - **Sin fondo:** Imprime una página sin su fondo.
  - **Color a negro:** Imprime todos los colores en negro para asegurar que todas las formas estarán visibles en el dibujo impreso. Esto resulta útil si una impresora

monocroma convierte los colores más suaves de un dibujo a color blanco, en lugar de a un tono de gris.

- **Imprimir a un archivo:** Envía la información para imprimirla en un archivo de un disco, en lugar de en la impresora. Al hacer clic en *Aceptar*, se abrirá el cuadro de diálogo *Imprimir a un archivo*, en el que puede proporcionar un nombre para el archivo de impresión y especificar dónde se guardará. Puede enviar el archivo de impresión resultante desde cualquier equipo conectado a la impresora especificada para el archivo de dibujo.
- **Todas:** Imprime todas las páginas del archivo de dibujo en el orden en que están guardadas en el archivo. Ésta es la opción predeterminada.
- **Página actual:** Imprime la página de dibujo que se muestra en cada momento.
- **Páginas (Desde/hasta):** Imprime sólo el intervalo concreto de páginas identificado mediante los valores escritos en los cuadros *Desde* y *Hasta*.
- **Selección:** Imprime las formas seleccionadas en cada momento.
- **Vista actual:** Imprime la parte del dibujo que se muestra en la ventana del programa.
- **Ajustar a una hoja:** Reduce el área para ajustarla al tamaño del papel especificado en el cuadro de diálogo *Configurar página*.
- **Número de copias:** Especifica el número de copias que se imprimirán (el valor predeterminado es 1).





*Figura 28: Vista de la opción Imprimir*

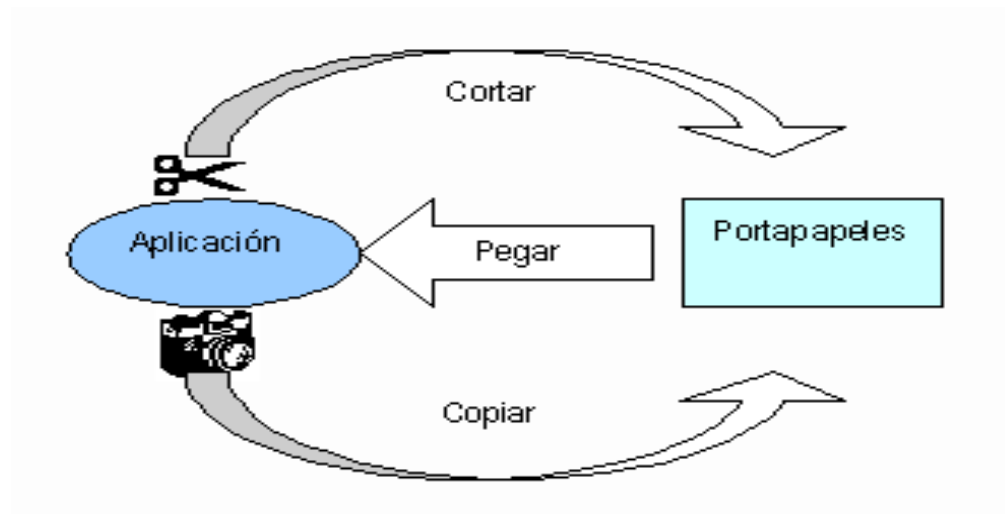
- **Cerrar:** Esta opción se utiliza para cerrar el diagrama activo en ese momento. Si se ha realizado alguna modificación en el diagrama que no esté guardada, nos dará la opción de hacerlo antes de cerrar. Esta opción no conlleva el cierre de la aplicación, únicamente del diagrama seleccionado.
- **Salir:** Esta opción cierra todos los diagramas abiertos en ese momento y cierra la sesión de la aplicación. Si se ha realizado alguna modificación que no esté guardada en alguno de los diagramas que tengamos abiertos, se dará la opción de salvarla antes de cerrar.

### ➤ Edición

- **Copiar:** Esta opción copia un objeto del diagrama y lo lleva al portapapeles de Windows. Previamente hay que seleccionar el objeto. Esta acción, al contrario que la de cortar, donde desaparece la selección, da la sensación de no hacer nada, porque el diagrama sigue igual. Posee también atajo de teclado, para ello debemos pulsar las teclas Ctrl + C.

- **Pegar:** Realiza el proceso contrario a las otras dos opciones anteriores, poniendo el contenido del portapapeles en la aplicación. El proceso de pegar no vacía el contenido del portapapeles, permaneciendo dicho contenido hasta que de nuevo se utilice la opción cortar o copiar y reemplace el contenido actual del portapapeles. Para ejecutar esta acción, previamente debemos haber utilizado alguna de las dos opciones anteriores antes. Si la forma o formas que vamos a pegar tenía datos definidos, estos se mantendrán. Tiene atajo de teclado, Ctrl + V.

\*Nota: El portapapeles es una zona de memoria auxiliar donde se almacena temporalmente información, accesible desde la mayoría de los programas de Windows. En la siguiente figura vemos su funcionamiento:



*Figura 29: Funcionamiento del portapapeles.*

- **Seleccionar todo:** Esta opción selecciona todos los objetos del diagrama actual. Todas las formas aparecerán como seleccionadas, ya que su contorno cambia de color, y además, aparecerá un cuadro rodeándolas, el cual nos ayuda a ver qué figuras entran dentro de la selección. También será seleccionado el texto si lo hubiera en el diagrama.

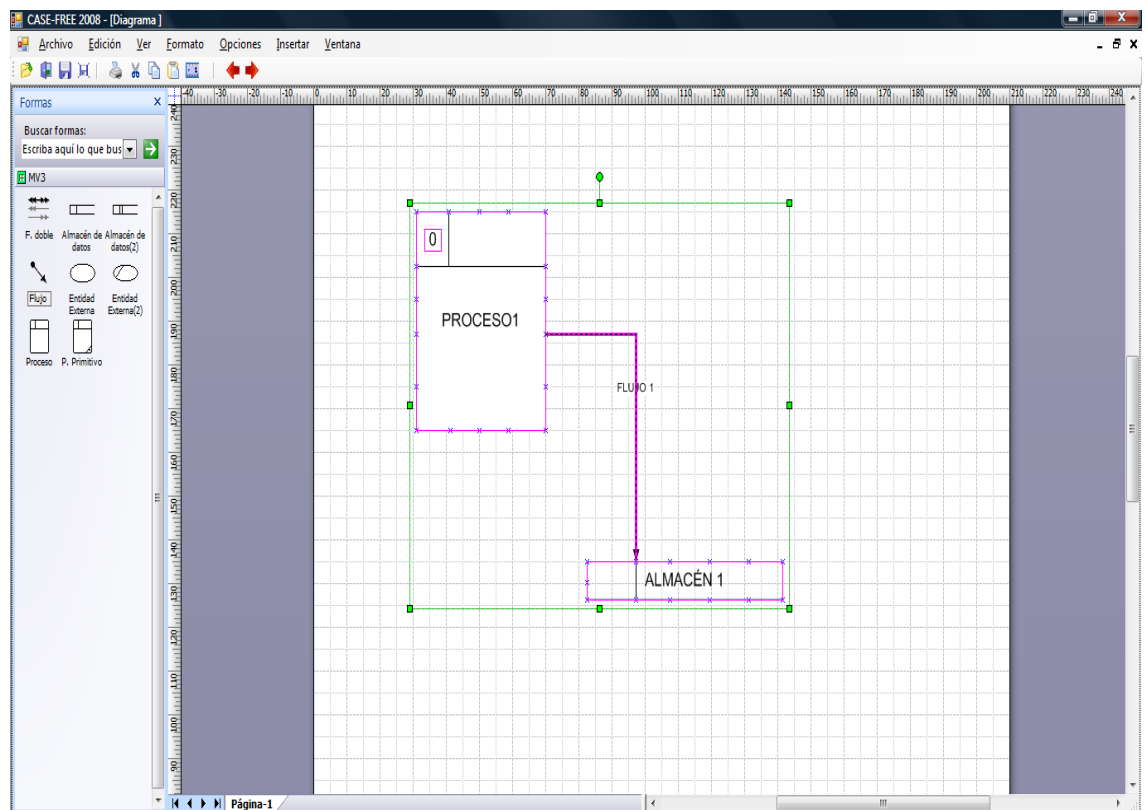


Figura 30: Vista de la opción Seleccionar todo

- **Buscar:** Esta opción permitirá buscar texto en nuestro diagrama. Se abrirá un cuadro de diálogo con las siguientes opciones:
  - **Buscar:** Permite especificar el texto que se desea buscar. Se puede escribir o pegar texto en este campo. Para repetir una búsqueda anterior, haremos clic en la flecha abajo y, a continuación, elegiremos una búsqueda de la lista.
  - **Especial:** Muestra una lista de caracteres especiales que se pueden buscar. Si se hace clic en un carácter especial de la lista, éste se agrega al cuadro Buscar.
  - **Selección:** La búsqueda se realiza únicamente en los elementos que tengamos seleccionados en ese momento.
  - **Página actual:** La búsqueda se realiza únicamente en el nivel actual.
  - **Todas las páginas:** La búsqueda se realiza en todos los niveles del diagrama.
  - **Texto de forma:** La búsqueda se realiza en los textos de las figuras o textos que tenga el diagrama.
  - **Datos de formas:** La búsqueda se realiza en el texto almacenado en los datos de la forma.
  - **Nombre de forma:** La búsqueda se realiza en los nombres de las formas (el nombre que se ve debajo de una forma en la paleta de figuras).

- **Celda definida por el usuario:** No aplica.
- **Coincidir mayúsculas y minúsculas:** Especifica que todas las coincidencias que se detecten deben tener exactamente la combinación de mayúsculas y minúsculas indicada en el cuadro Buscar.
- **Caracteres del mismo ancho:** Especifica que todas las coincidencias que se detecten deben tener exactamente el ancho de carácter indicado en el cuadro Buscar. Por ejemplo, el alfabeto Katakana se puede representar mediante caracteres anchos o estrechos.
- **Sólo palabras completas:** Especifica que todas las coincidencias que se detecten deben ser palabras completas y no partes de palabras más largas. Por ejemplo, si se escribe por en el cuadro Buscar, se encontrarán todas las apariciones de "por", pero "portal" se pasará por alto.
- **Encontrado en:** Determina si el texto coincidente se ha encontrado en un bloque de texto, o muestra el nombre de la forma o en los datos de la forma.
- **Buscar Siguiente:** Se busca la siguiente aparición del texto del cuadro *Buscar*.

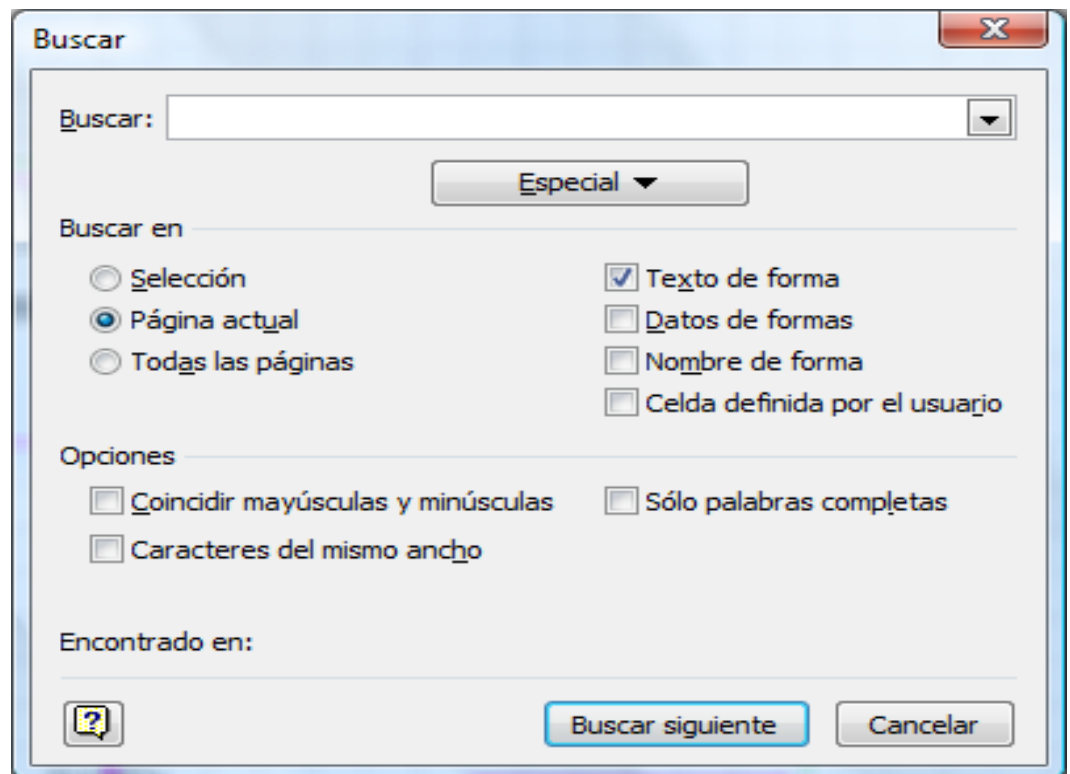
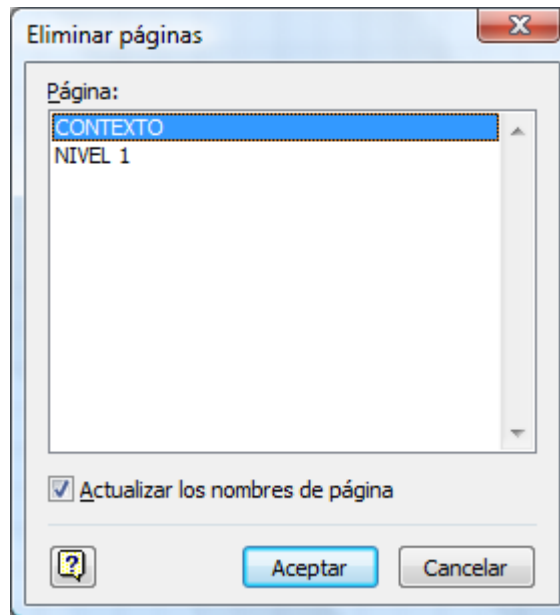


Figura 31: Vista de la opción Buscar

- **Eliminar Páginas:** Esta opción nos permitirá eliminar una página o nivel del diagrama actual. Se nos abrirá una ventana emergente en la cual podremos seleccionar la página o páginas que queremos eliminar, así como la opción de que se

actualicen los nombres de las páginas automáticamente (sólo se aplica a las páginas que utilizan nombres predeterminados, como Página-1.). Si eliminamos todas las páginas que tenemos, se nos abrirá una nueva en blanco.



*Figura 32: Vista de la opción Eliminar páginas*

➤ **Ver**

- **Zoom:** Esta opción nos permitirá ver una panorámica del nivel actual así como ampliar o disminuir su visualización. Nos abrirá una ventana emergente que podremos acoplar en el lugar que mejor nos convenga de la aplicación. Dispondrá de hasta diez niveles de zoom para aumentar o disminuir la vista. Dentro de la venta emergente, dispondremos de la opción Activar Ocultar automáticamente, con la cual, la ventana se activará si ponemos por encima el puntero del ratón o se ocultará cuando lo quitemos. Para cerrarla basta con pulsar sobre el aspa o de nuevo pulsar sobre la opción del menú. Esta opción es común para todos los niveles del diagrama, es decir, si la mostramos se mostrará en todos los niveles, y si la ocultamos, se ocultará también en todos.

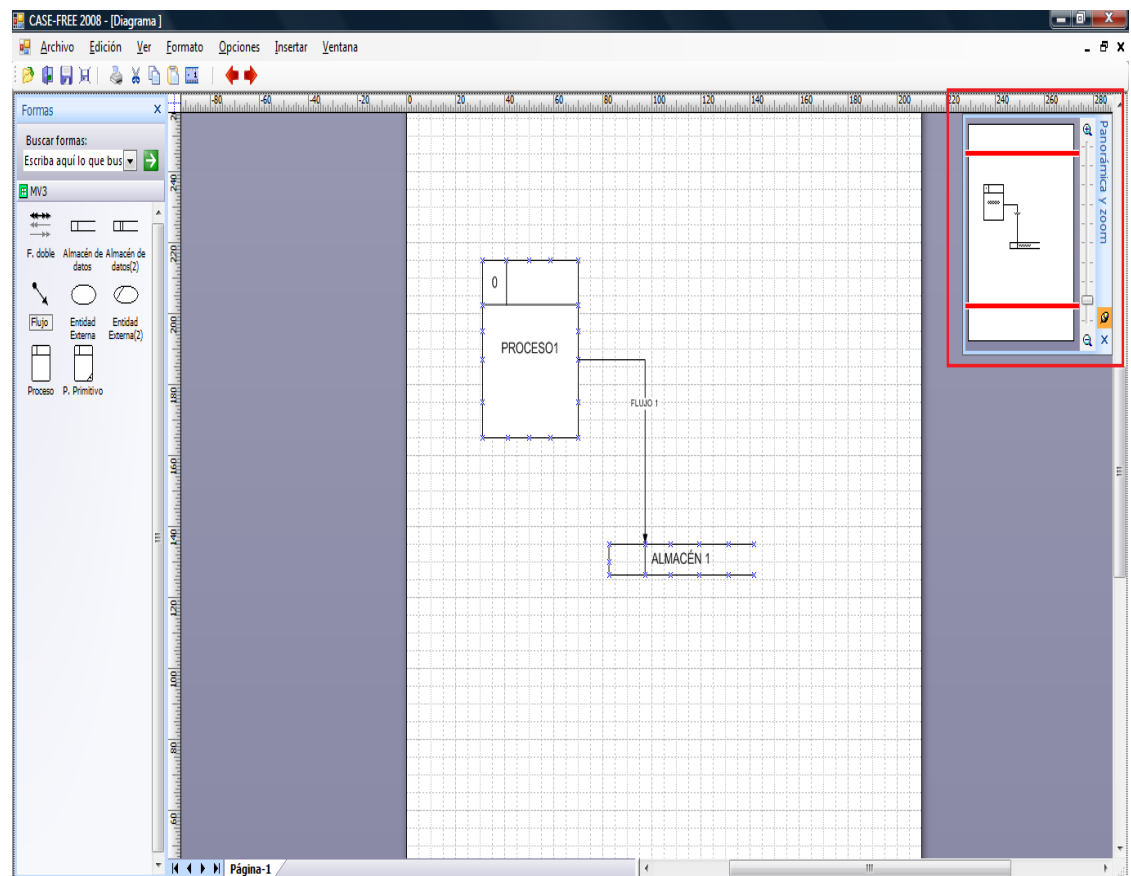


Figura 33: Vista de la opción Zoom

- **Propiedades:** Esta opción nos permitirá agregar o modificar valores a los campos de datos de las figuras. Para ellos nos abrirá una ventana emergente que podremos acoplar en el lugar que mejor nos convenga, e iremos agregando los valores en los campos de datos de las formas seleccionadas. Esto debemos hacerlo seleccionando las figuras de una en una. Para cerrar la ventana basta con pulsar sobre el aspa o de nuevo pulsar sobre la opción del menú.

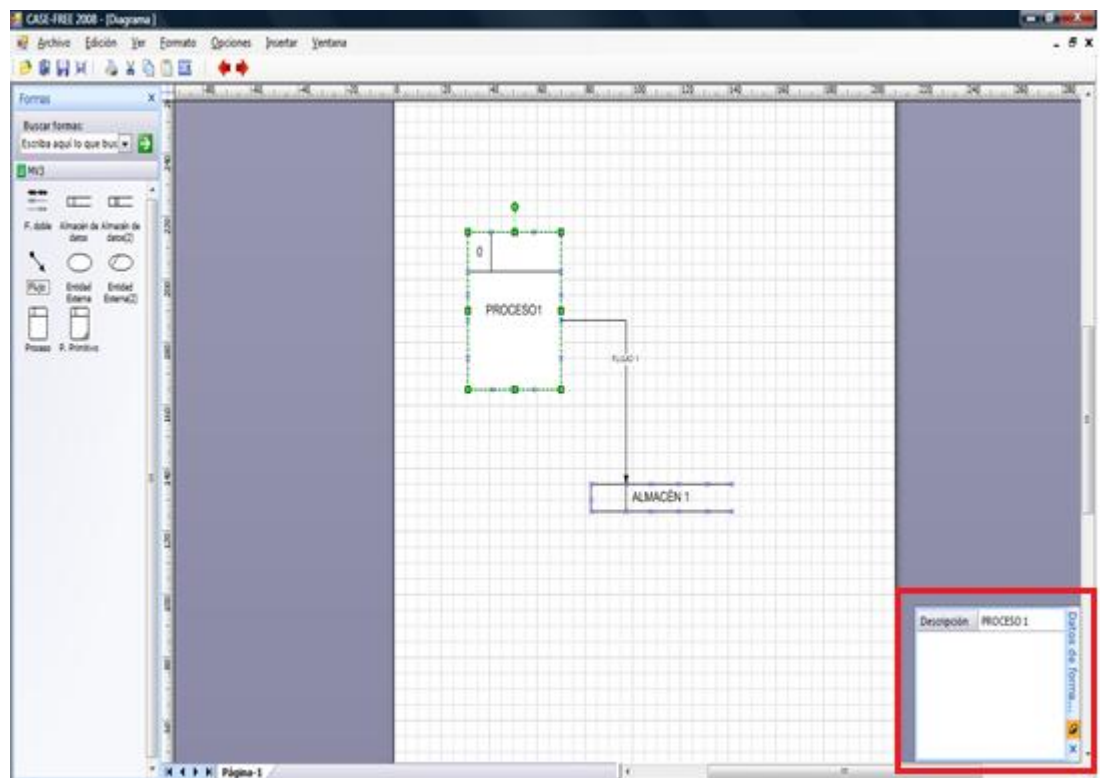


Figura 34: Vista de la opción *Propiedades*

Si por el contrario lo que queremos es agregar campos de datos a las formas, debemos hacer clic con el botón derecho del ratón sobre una figura, después entramos en Datos → Datos de formas. Para asignarle un nombre al campo, pulsaremos el botón definir, donde tenemos las siguientes opciones:

- **Etiqueta:** Especifica el nombre del dato. La etiqueta aparecerá junto al campo donde escribiremos los datos. Una etiqueta está compuesta de caracteres alfanuméricos, incluyendo el carácter de subrayado (\_).
- **Tipo:** Especifica el tipo de datos para el valor que vamos a introducir, que son los siguientes:
  - ❖ **Cadena:** Datos alfanuméricos.
  - ❖ **Número:** Datos numéricos.
  - ❖ **Lista fija:** No procede.
  - ❖ **Lista variable:** No procede.
  - ❖ **Booleano:** Datos booleanos (Verdadero/Falso).
  - ❖ **Moneda:** Datos de moneda en los que se emplea la configuración regional de Microsoft Windows para los decimales y los símbolos de moneda.
  - ❖ **Fecha:** Fechas y horas. El tipo de calendario se define en el cuadro *Calendario* y afecta a las opciones de formato de fecha disponibles en la lista *Formato*.

- ❖ **Duración:** Periodo de existencia en el tiempo.
- ❖ **Idioma:** Identifica el idioma asociado a los tipos de datos Fecha y Cadena.
- **Formato:** Especifica el modo en que se muestran los datos especificados. La forma en que se usa este campo depende de la configuración de las opciones Tipo y Calendario. En el caso de los tipos de datos de cadena, numéricos, de moneda, de fecha y de duración, seleccionaremos uno de los formatos disponibles en el cuadro formato. Esta opción no estará disponible para tipos de datos booleanos.
- **Calendario:** Establece el tipo de calendario que se debe emplear para el idioma seleccionado: hijri árabe (islámico), lunar hebreo (judío), saka (hindú) u occidental (gregoriano). El tipo de calendario afectará a las opciones disponibles en la lista Formato. Cuando escribamos una fecha, se convertirá en una fecha del calendario seleccionado.
- **Valor:** Contiene el valor inicial del dato.
- **Petición de datos:** Especifica los datos que se nos van a pedir cuando ejecutemos la opción de *Propiedades* en nuestro menú o al hacer clic con el botón derecho y entrar en Datos → Datos de formas. Todas las figuras de la paleta vienen ya con datos predefinidos.
- **Lista de propiedades:** Enumera todas las propiedades definidas para la forma seleccionada. Podremos seleccionar una propiedad para modificarla o eliminarla.
- **Nueva:** Agrega una nueva propiedad a la lista.
- **Eliminar:** Elimina la propiedad seleccionada.



**Definir datos de formas**

Etiqueta:

Tipo:  Idioma:

Formato:  Calendario:

Valor:

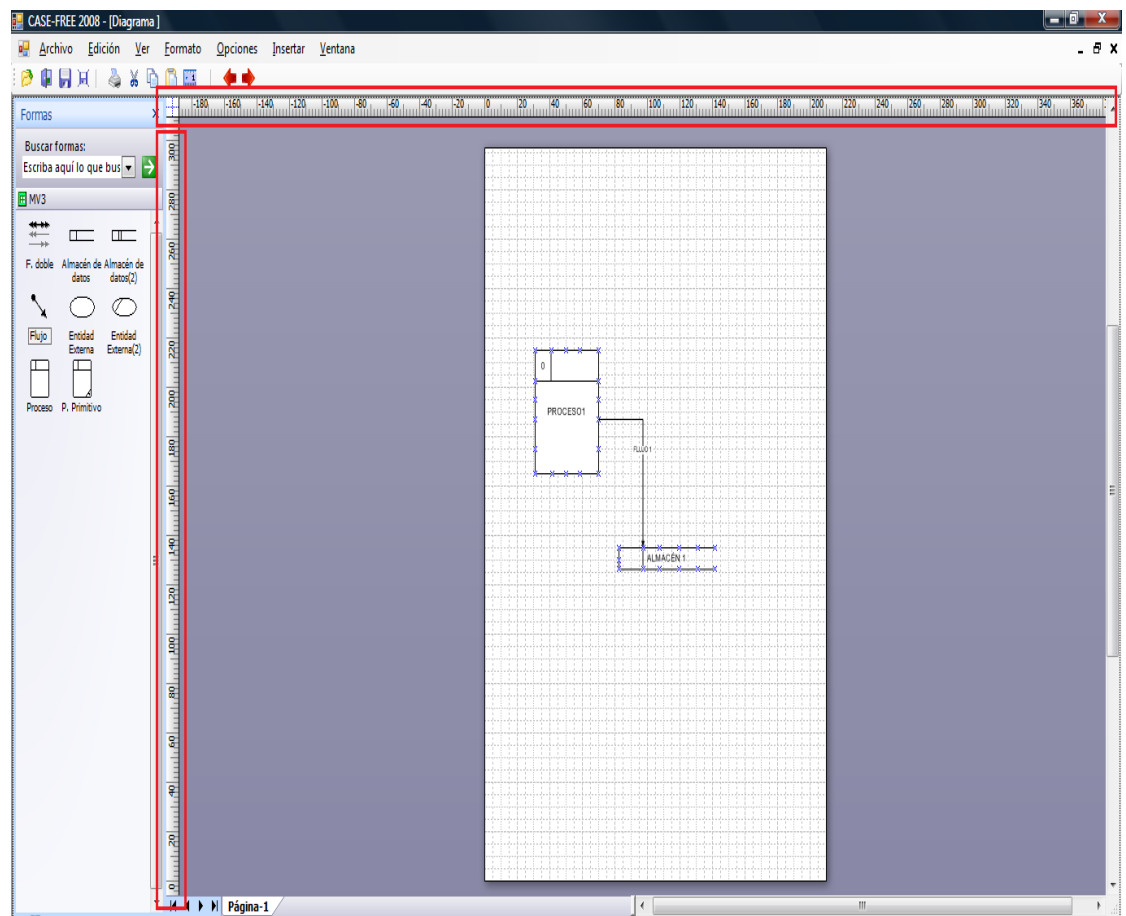
Petición de datos:

Propiedades:

Etiqueta	Tipo	Formato
Dato1	Cadena	
Dato2	Cadena	
Dato3	Cadena	
Dato4	Cadena	
Dato5	Cadena	

Figura 35: Vista Definir datos de formas

- **Regla:** Con esta opción ocultaremos o pondremos visible las regletas que están situadas en la parte izquierda y superior del diagrama. Por defecto la unidad de medida que aparecen son milímetros, pero podremos cambiarlo con la opción de *Configurar página*, dentro de la pestaña *Propiedades* de la página, en el desplegable *Unidades de medida*.



*Figura 36: Vista de las reglas*

- **Formas:** Con esta opción, si hemos cerrado la paleta de objetos que se encuentra en la parte izquierda del aplicativo, la volveremos a hacer visible. Para situar un objeto de la paleta dentro del diagrama, debemos seleccionarla con el ratón, y con el botón izquierdo pulsado, la arrastraremos hasta el lugar que deseemos.

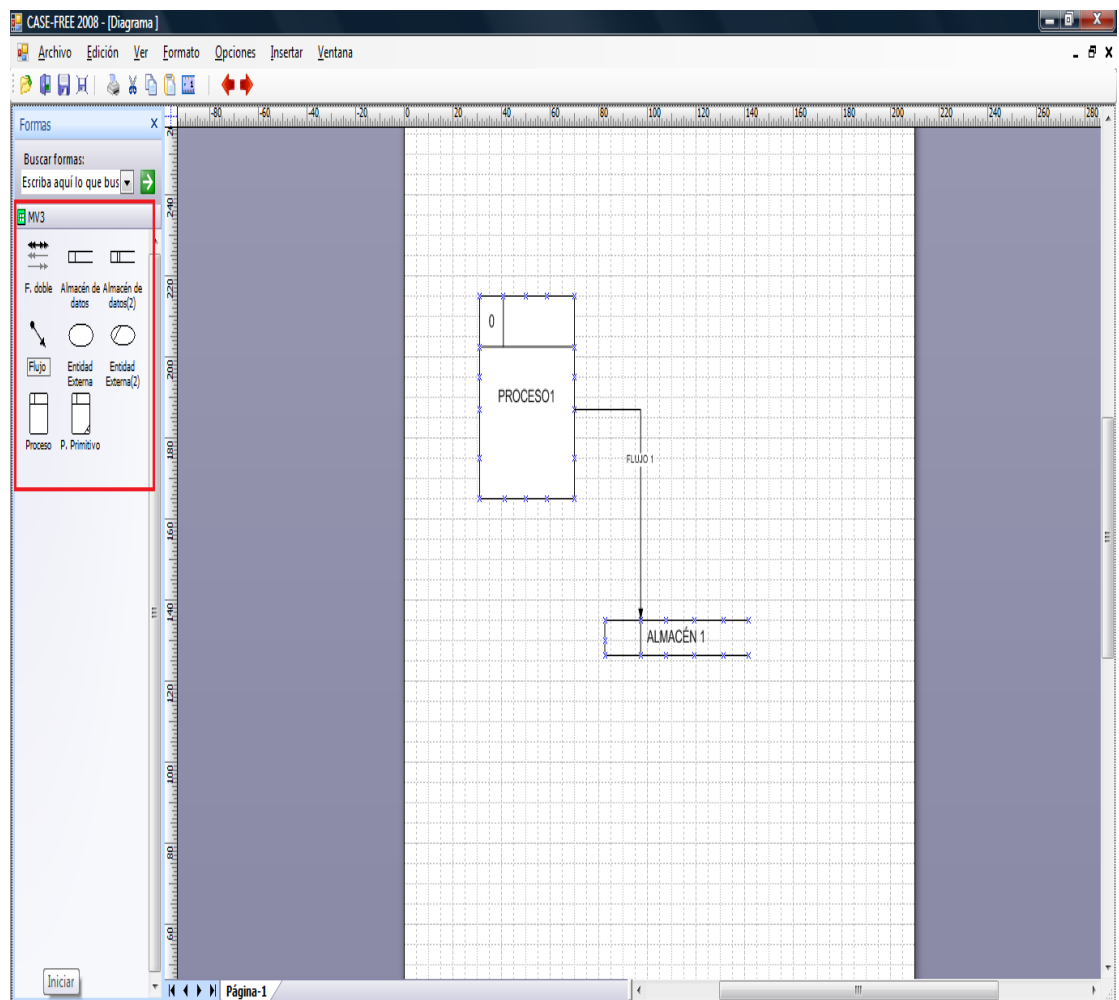
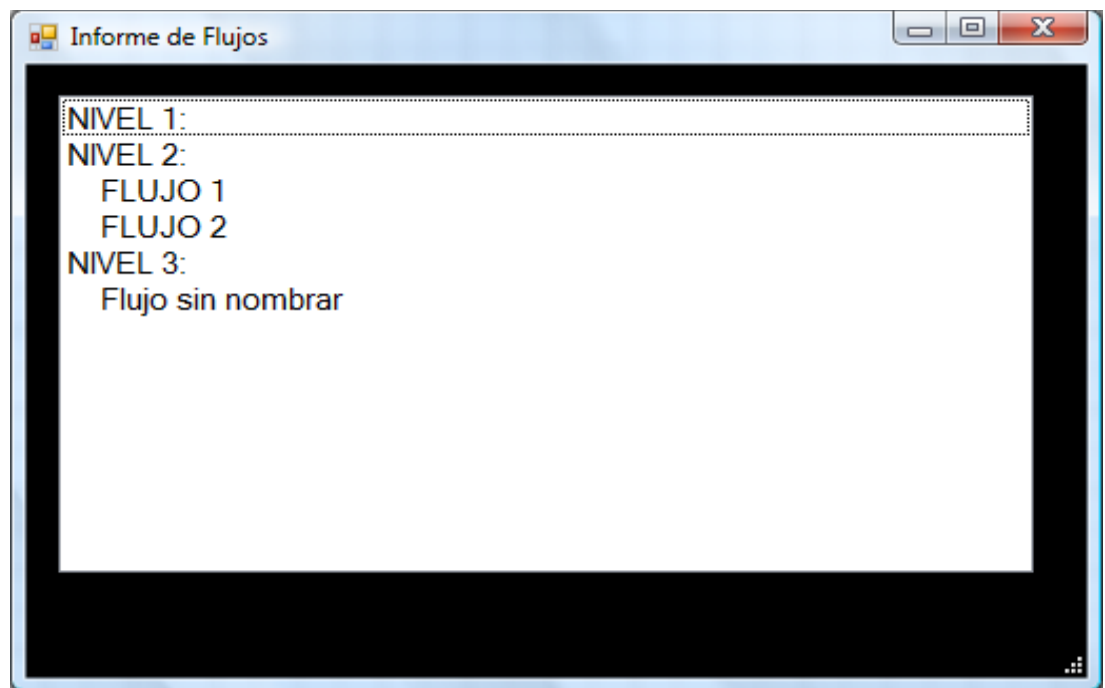


Figura 37: Vista de la paleta de formas

- **Flujos Actuales:** Esta opción sacará una ventana emergente con el nombre de todos los flujos que se encuentran en el diagrama, separados por niveles. Si el flujo no tiene nombre, se mostrará el literal “Flujo sin nombrar”. Mostrará un mensaje de aviso si no encontrara flujos en ningún nivel. Si un nivel no tiene flujos, lo dejará en blanco. Para cerrar esta ventana, pulsaremos sobre el aspa.



*Figura 38: Vista Informe de flujos*

- **Informe:** Esta opción nos permitirá generar un informe con todas las figuras de la página actual o de todo el diagrama. Se nos abrirá un cuadro de diálogo con las siguientes opciones:
  - **Nuevo:** Inicia el asistente para definición de informes, con el fin de que podamos crear un informe nuevo.
  - **Modificar:** Inicia el asistente para definición de informes, con el fin de que podamos cambiar la definición del informe que seleccionemos.
  - **Quitar:** Elimina de la lista el informe seleccionado. Sólo podremos eliminar los informes que estén guardados en el diagrama. Para quitar informes que estén guardados en archivos, debemos borrar esos archivos.
  - **Examinar:** Permite buscar un informe almacenado en un archivo que no se encuentra en ninguna de las ubicaciones de búsqueda predeterminadas.
  - **Descripción:** Describe la descripción del informe seleccionado.
  - **Mostrar sólo los informes específicos del dibujo:** Indica si la lista de definiciones de informes se limita a los informes que sean pertinentes para el diagrama abierto. Si se desactiva esta opción, la lista incluye todas las definiciones de informes.
  - **Ejecutar:** Abre el cuadro de diálogo *Ejecutar informe*, donde podremos seleccionar el formato para el informe y crear un informe basado en la definición seleccionada. La opción guardar como solo estará disponible para formatos html o

xml, y podremos guardar el informe en la ruta que queramos. Si seleccionamos el formato forma de visio tendremos dos opciones:

- ❖ **Copia de la definición del informe:** Almacena una copia de la definición del informe con la forma que contiene el informe generado. Los cambios que realicemos en la definición del informe original no cambiarán la forma del informe.
- ❖ **Vincular a la definición del informe.** Vincula la forma que contiene el informe generado a una definición de informe. Los cambios que realicemos a la definición del informe afectarán al informe almacenado en la forma la próxima vez que actualice el informe.

Cuando estemos generando el informe, podremos seleccionar las propiedades de cada figura que deseemos que se muestren, como el alto, ancho, nombre, ubicación..., ordenarlas por algún criterio, mostrar subtotales de las propiedades, darles un formato o guardar una descripción. Tanto si creamos un informe nuevo, como si lo modificamos, en el asistente tendremos un botón de opciones avanzadas, que nos sirve para especificar los criterios que deben cumplir las formas para que se incluyan en el informe que vamos a generar. Tendremos las siguientes opciones:

- ❖ **Propiedad:** Especifica una propiedad que se va a incluir en el criterio de selección. Las propiedades incluyen datos de formas, nombres de capas, las propiedades definidas por el usuario y formas de Autodiscovery (no aplica).
- ❖ **Condición:** Especifica una restricción o cualificación que debe cumplir la propiedad. Las condiciones varían en función de la propiedad seleccionada.

Condición	Acción	Se utiliza para
>	Mayor que	Propiedades cuyo tipo es número
<	Menor que	Propiedades cuyo tipo es número
>=	Mayor o igual que	Propiedades cuyo tipo es número
<=	Menor o igual que	Propiedades cuyo tipo es número
=	Igual a	Propiedades de cualquier tipo
<>	Distinto de	Propiedades de cualquier tipo

Tabla 43: Condiciones para los informes

- ❖ **Valor:** Especifica un valor que completa la condición, como texto o un número. El valor que podemos especificar depende de la propiedad y la condición seleccionadas. Si seleccionamos datos de formas cuyo tipo es *Fecha*, debemos poner una fecha válida utilizando el formato especificado para la propiedad.
- ❖ **Criterios definidos:** Muestra una lista con los criterios incluidos en la expresión de selección.
- ❖ **Agregar:** Agrega el criterio especificado en los cuadros *Propiedad*, *Condición* y *Valor* a la lista *Criterios definidos*.
- ❖ **Eliminar:** Quita de la expresión de selección los criterios seleccionados en la lista *Criterios definidos*.
- ❖ **Borrar:** Elimina de la expresión de selección todos los criterios de la lista *Criterios definidos*.
- ❖ **Distinguir mayúsculas de minúsculas:** Indica que el criterio debe distinguir entre mayúsculas y minúsculas.

Para guardarlo, podremos hacerlo sobre el mismo dibujo, o en un archivo diferente.

Una vez generado el informe, podremos visualizarlo en cuatro formatos diferentes, excel, html, xml y formas de visio.

### Informe\_1

Id. de forma	Nombre de forma	Texto mostrado
6	Proceso	PROCESO 1
1	Flujo	FLUJO

*Figura 39: Informe en formato HTML*

#### ➤ Formato

- **Texto:** Esta opción abre una caja de diálogo para dar formato al texto que aparece en el diagrama. Estará dividida en cinco pestañas diferentes. Vemos las propiedades de cada una de ellas:
  - **Fuente:** Especifica las características de fuente y estilo del texto que seleccionemos o de todo el bloque de texto de la figura que seleccionemos. Sus opciones son:
    - ❖ **Fuente:** Muestra una lista de todas las fuentes instaladas en el equipo. Para cambiar una, la seleccionaremos de la lista.
    - ❖ **Estilo:** Especifica el estilo que se aplica al texto seleccionado (normal, cursiva, negrita o cursiva negrita).
    - ❖ **Tamaño:** Establece el tamaño de punto del texto seleccionado. El porcentaje se basa en el valor de tamaño del texto latino y asiático.
    - ❖ **Mayúsculas y minúsculas:** Especifica el formato de mayúsculas y minúsculas del texto:
      - **Normal:** Muestra el texto tal y como se escribe.
      - **Todas mayúsculas:** Muestra todos los caracteres del texto en mayúsculas.
      - **Primera en mayúscula:** Muestra el primer carácter de cada palabra en mayúsculas y los demás en minúsculas.

- **Versales:** Muestra todos los caracteres en minúsculas como letras versales (mayúsculas con un tamaño de punto menor que el especificado) y deja los caracteres en mayúsculas sin modificar.
  - ❖ **Posición:** Especifica la posición del texto:
    - **Normal:** Alinea el texto seleccionado horizontalmente en la línea base.
    - **Superíndice:** Sube el texto seleccionado ligeramente con respecto a la línea base y reduce su tamaño de punto.
    - **Subíndice:** Baja el texto seleccionado ligeramente con respecto a la línea base y reduce su tamaño de punto.
  - ❖ **Subrayado:** Dibuja una línea bajo el texto seleccionado:
    - **(ninguno):** Quita el subrayado del texto seleccionado.
    - **Simple:** Dibuja una sola línea bajo el texto seleccionado.
    - **Doble:** Dibuja una línea doble bajo el texto seleccionado.
  - ❖ **Tachado:** Dibuja una línea a través del centro del texto.
    - **(ninguno):** Quita el tachado del texto seleccionado.
    - **Simple:** Dibuja una línea a través del texto seleccionado.
    - **Doble:** Dibuja una línea doble a través del texto seleccionado.
  - ❖ **Color:** Establece el color del texto.
  - ❖ **Idioma:** Especifica la configuración del idioma.
  - ❖ **Transparencia:** Especifica en qué proporción es opaco o transparente el texto. El cero por ciento es completamente opaco y el 100 por ciento es completamente transparente.
- **Carácter:** Establece la cantidad de espacio que debe haber entre dos o más caracteres y estrecha o amplía los propios caracteres. Sus opciones son:
    - ❖ **Escala:** Estrecha los caracteres seleccionados cuando se especifica un porcentaje menor que 100 por ciento, o los amplía cuando se especifica un porcentaje mayor que 100 por ciento. Puede seleccionar un valor de la lista o escribir uno entre 1 y 600 por ciento.
    - ❖ **Espaciado:** Determina cómo aplicar espacio entre los caracteres seleccionados. La configuración predeterminada es *Normal*. *Expandido* separa las letras. *Comprimido* acerca las letras.
      - **En:** Establece el espacio que se alejarán o aproximarán los caracteres seleccionados. Podemos escribir un valor en puntos entre +1584 (expandido) y -1584 (condensado). Un número negativo muy grande puede hacer que las letras se superpongan.



- **Párrafo:** Especifica la alineación, la sangría y el espaciado de los párrafos de texto seleccionados o de todo el de la forma seleccionada. Sus opciones son:
  - ❖ **Alineación horizontal:** Muestra la alineación horizontal del texto en relación con el margen del bloque de texto.
    - **Izquierda:** Inicia cada línea de texto en el margen izquierdo. El lado derecho del texto será desigual.
    - **Centrado:** Centra cada línea de texto entre los márgenes izquierdo y derecho.
    - **Derecha:** Coloca el último carácter de cada línea en el margen derecho. El lado izquierdo del texto será desigual.
    - **Justificado:** Ajusta el espaciado entre palabras y caracteres para que cada línea de texto (excepto la última línea del párrafo) rellene el espacio que hay entre los márgenes izquierdo y derecho.
  - ❖ **Sangría:** Si no hay selección de texto, se aplicará a todo el bloque de texto.
    - **Antes del texto:** Establece la sangría de los párrafos desde el margen inicial. En un párrafo de izquierda a derecha, desde el margen izquierdo y, en un párrafo de derecha a izquierda, desde el margen derecho.
    - **Después del texto:** Establece la sangría de los párrafos desde el margen final. En un párrafo de izquierda a derecha, desde el margen derecho y, en un párrafo de derecha a izquierda, desde el margen izquierdo.
    - **Primera línea:** Establece la sangría de la primera línea de texto en relación con la sangría inicial.
  - ❖ **Espaciado:** Muestra el espaciado entre párrafos y líneas del párrafo seleccionado o de todo el bloque de texto si no hay ningún párrafo seleccionado.
    - **Antes:** Especifica el espacio que se inserta antes de un párrafo (excepto para el primer párrafo del bloque de texto)
    - **Después:** Especifica el espacio que se inserta después de un párrafo (excepto para el último párrafo del bloque de texto). Si especificó un valor en Antes, la cantidad de espacio entre párrafos será igual a la suma de los valores de Antes y Después.
    - **Línea:** Especifica el espacio entre las líneas de un párrafo.
- **Bloque de texto:** Establece la alineación vertical, los márgenes y el color de fondo del bloque de texto. Sus opciones son:

- ❖ **Alineación:** Especifica la posición vertical del texto con respecto a los márgenes superior e inferior del bloque de texto.
    - **Superior:** Alinea la línea superior del texto con el margen superior.
    - **En medio:** Centra el texto entre los márgenes superior e inferior, de manera que la distancia entre la línea superior y el margen superior es igual a la distancia entre la línea inferior y el margen inferior.
    - **Inferior:** Alinea la línea inferior del texto con el margen inferior.
  - ❖ **Márgenes:**
    - **Superior:** Especifica la distancia mínima entre el texto y el borde superior del bloque de texto.
    - **Inferior:** Especifica la distancia mínima entre el texto y el borde inferior del bloque de texto.
    - **Izquierda:** Especifica la distancia mínima entre el texto y el borde izquierdo del bloque de texto.
    - **Derecha:** Especifica la distancia mínima entre el texto y el borde derecho del bloque de texto.
  - ❖ **Ninguno:** Especifica un fondo invisible (transparente) que permite ver las formas que se encuentran detrás del texto.
  - ❖ **Color sólido:** Especifica del color de fondo.
  - ❖ **Transparencia:** Especifica en qué proporción es opaco o transparente el texto. El cero por ciento es completamente opaco y el 100 por ciento es completamente transparente.
- **Tabulaciones:** Podremos agregar, eliminar y ajustar tabulaciones en los párrafos seleccionados o en todo el bloque de texto (hasta 60). Sus opciones son:
    - ❖ **Tabulaciones:** Enumera las tabulaciones actuales. El lado izquierdo del cuadro enumera las posiciones de tabulación y el lado derecho muestra la alineación de las tabulaciones.
    - ❖ **Posición de tabulación:** Especifica la posición de la tabulación seleccionada en la lista Tabulaciones.
    - ❖ **Alineación:**
      - **Izquierda:** Alinea el texto a la izquierda de la tabulación.
      - **Derecha:** Alinea el texto a la derecha de la tabulación.
      - **Centro:** Alinea el texto en el centro de la tabulación.
      - **Decimal:** Alinea el signo decimal de los números con el signo decimal de la tabulación. (Alinea el texto a la izquierda del signo decimal de la tabulación.) Si en la configuración de idioma en uso se emplea la coma

como separador decimal, el texto se alinea con respecto a la tabulación en la coma.

- ❖ **Tabulaciones predeterminadas:** Establece la distancia entre las tabulaciones predeterminadas para la forma seleccionada.
  - ❖ **Agregar:** Agrega una nueva tabulación, separada de la anterior por el espacio especificado en el cuadro Tabulaciones predeterminadas.
  - ❖ **Modificar:** Asigna a la tabulación seleccionada en la lista Tabulaciones el valor especificado en el cuadro Posición de tabulación.
  - ❖ **Quitar:** Quita la tabulación seleccionada en la lista Tabulaciones.
  - ❖ **Quitar todas:** Quita todas las tabulaciones de la forma seleccionada.
- 
- **Viñetas:** Esta ficha especifica las opciones de viñeta para los párrafos seleccionados en un bloque de texto o para todo el bloque de texto. Sus opciones son:
    - ❖ **Tamaño de fuente:** Especifica el tamaño de la viñeta, pero no afecta al resto del texto.
    - ❖ **Posición del texto:** Especifica el espacio entre la viñeta y el texto.
    - ❖ **Caracteres de viñeta:** Escriba uno o más caracteres para una viñeta personalizada.
    - ❖ **Fuente:** Especifica la fuente de los caracteres de una viñeta personalizada.

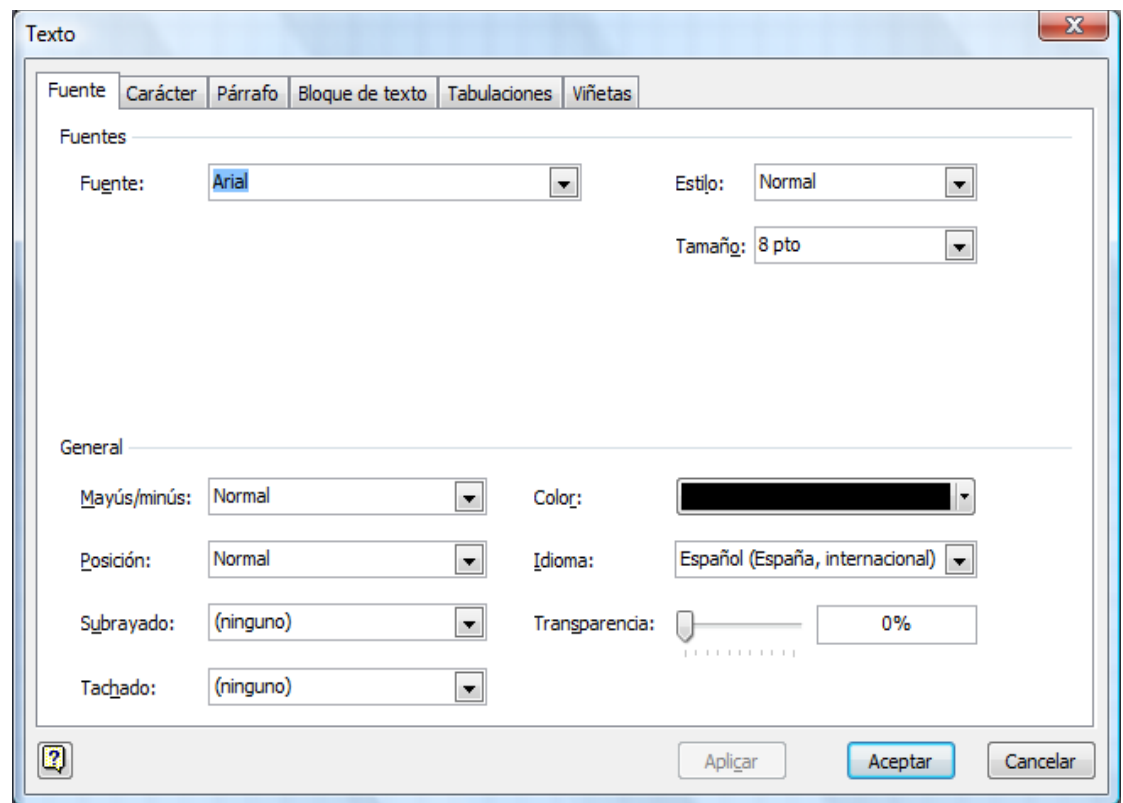


Figura 40: Vista de la opción Texto

- **Línea:** Las opciones de este cuadro de diálogo aplican tipos de formato como tramas, grosor y extremos a las líneas seleccionadas. Tendremos las siguientes opciones:
  - **Trama:** Tendremos una lista en la que podremos seleccionar ninguna, líneas sólidas, líneas con guiones o bien una trama mixta.
  - **Peso:** En la lista, podremos seleccionar el grosor de línea que deseemos o especificar un grosor de línea personalizado.
  - **Color:** Podremos seleccionar un color entre los disponibles en la paleta. Si queremos agregar un nuevo color a la paleta, haremos clic en la opción *Más colores* y, a continuación, en el cuadro de diálogo *Colores*, definiremos el nuevo color.
  - **Remate:** Seleccionaremos remates de línea redondos o cuadrados para que el extremo de una línea aparezca redondeado o cuadrado. Esta opción sólo estará disponible para líneas muy gruesas.
  - **Transparencia:** Con el control deslizante podremos configurar, de 0 a 100, la transparencia de la línea seleccionada.
  - **Inicio:** Podremos seleccionar el extremo de línea que deseemos para el punto de inicio de los flujos.

- **Fin:** Podremos seleccionar el extremo de línea que deseemos para el punto final de los flujos.
- **Tamaño inicial:** Especifica el tamaño del punto inicial de los flujos.
- **Tamaño final:** Especifica el tamaño del punto final de los flujos.
- **Redondeo:** Redondea las esquinas de la forma seleccionada; cuanto mayor sea el número, más redondeadas serán las esquinas. Podremos elegir una de las esquinas redondeadas de los diseños escribir un valor (hasta 1) para especificar una esquina personalizada.
- **Vista previa:** Podremos visualizar una vista previa antes de aplicar los cambios

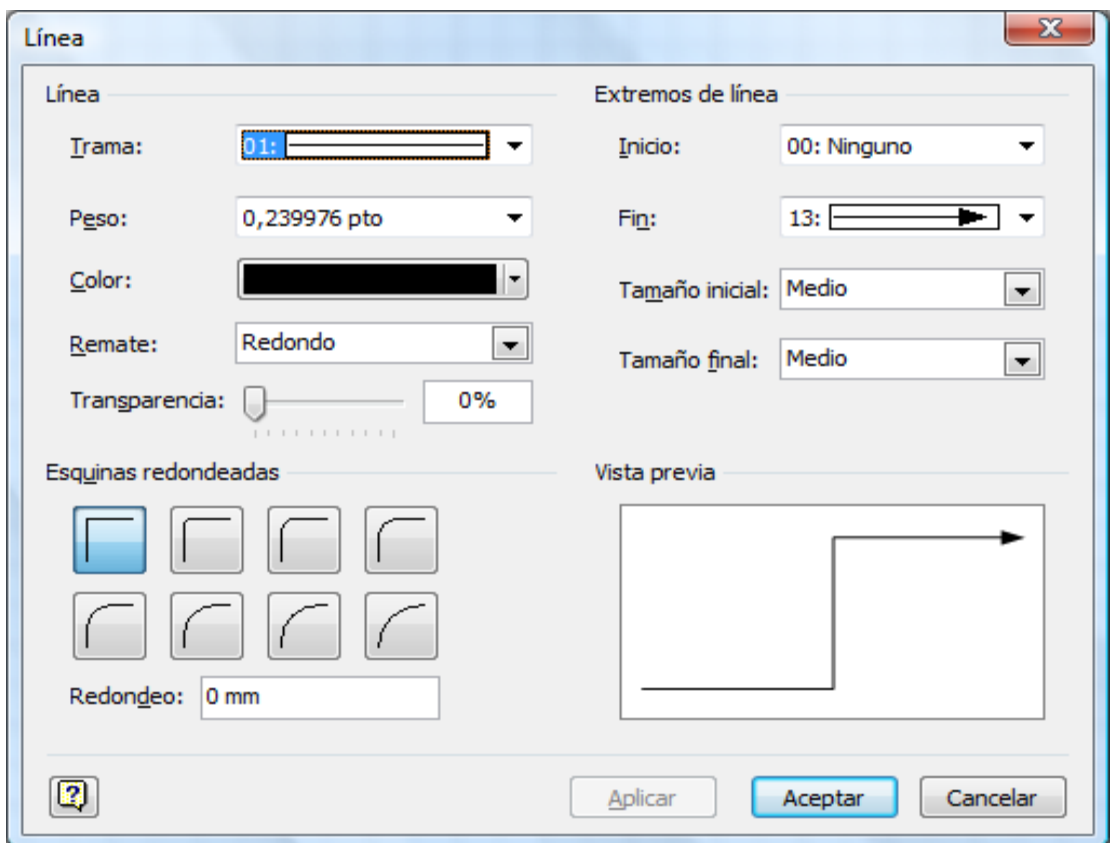


Figura 41: Vista de la opción Línea

- **Relleno:** Esta opción nos permite dar formato al relleno y aplicar sombras a las figuras del diagrama. Tendremos las siguientes opciones:
  - **Color(Relleno):** Podremos seleccionar un color entre los disponibles en la paleta. Si queremos agregar un nuevo color a la paleta, haremos clic en la opción *Más*

*colores* y, a continuación, en el cuadro de diálogo *Colores*, definiremos el nuevo color.

- **Trama(Relleno):** Muestra la trama de relleno actual para la figura seleccionada. Seleccionaremos una nueva trama, que podrá ser blanca o transparente (Ninguna o trama 0), Sólida (trama 1), una trama de mapa de bits (tramas 2 a 24) o una trama degradada (tramas 25 a 40).
- **Color de trama(Relleno):** Muestra el color de primer plano de la forma seleccionada. Seleccionaremos en la lista un nuevo o seleccionaremos la opción de *Más colores* para seleccionar un color personalizado en el cuadro de diálogo *Colores*.
- **Transparencia(Relleno):** Con el control deslizante podremos configurar, de 0 a 100, la transparencia de la trama del relleno. Si ponemos 100 por ciento de transparencia a todo, la figura será transparente.
- **Estilo:** Establece el estilo de la sombra a la figura seleccionada.
- **Color (Sombra):** Muestra el color de fondo que aparecerá detrás de los puntos y las líneas que componen la sombra de la forma seleccionada. Seleccionaremos en la lista un nuevo color de fondo. Si queremos agregar un nuevo color a la paleta, haremos clic en la opción *Más colores* y, a continuación, en el cuadro de diálogo *Colores*, definiremos el nuevo color.
- **Trama (Sombra):** Muestra la trama de la sombra de la forma seleccionada. Seleccionaremos una nueva trama, que podrá ser blanca o transparente (Ninguna o trama 0), Sólida (trama 1) o una trama de mapa de bits (tramas 2 a 24). Si seleccionamos *Ninguna* en la lista Trama, se eliminará la sombra. Al seleccionar *Ninguna*, no podremos aplicar las opciones *Color* o *Color de trama*.
- **Color de trama (Sombra):** Muestra el color de primer plano de los puntos y las líneas que componen la sombra de la figura seleccionada. Si queremos agregar un nuevo color a la paleta, haremos clic en la opción *Más colores* y, a continuación, en el cuadro de diálogo *Colores*, definiremos el nuevo color.
- **Transparencia:** Con el control deslizante podremos configurar, de 0 a 100, la transparencia de la sombra seleccionada.
  - ❖ **Vista previa:** Muestra una vista previa de las selecciones que hagamos de las sombras y el relleno.

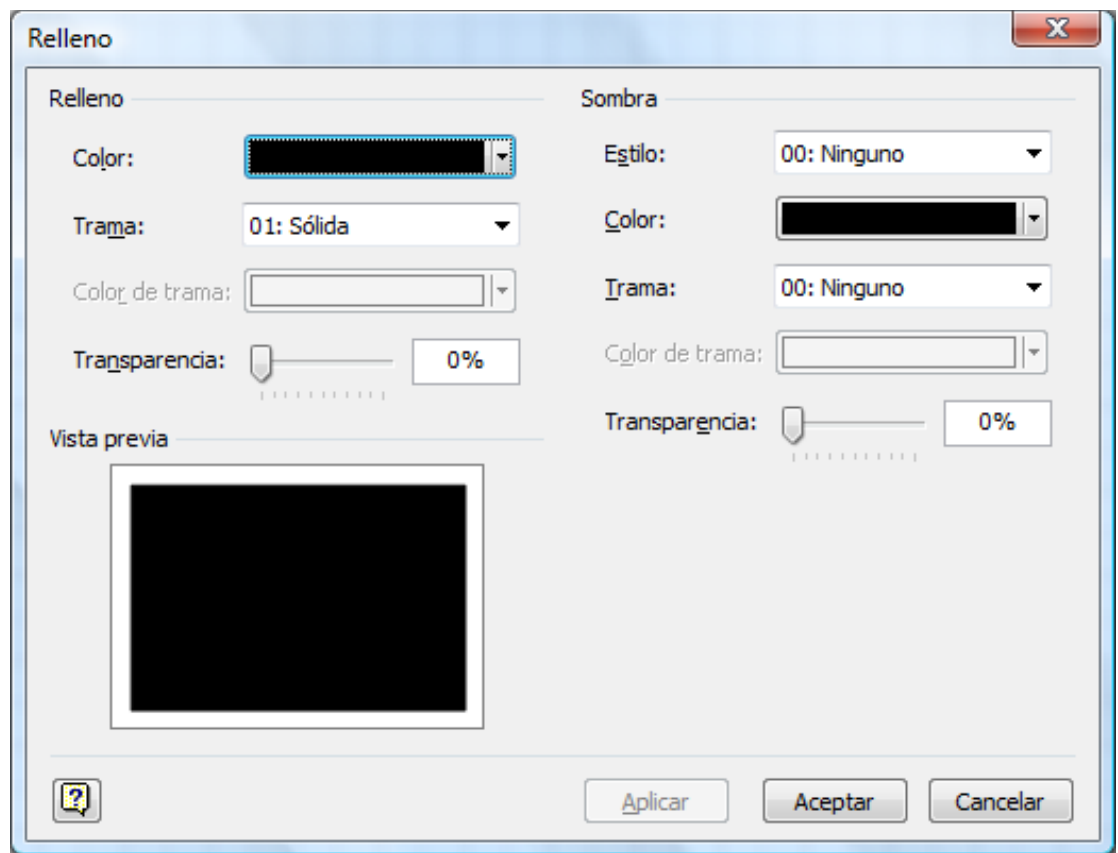


Figura 42: Vista de la opción Relleno

- **Sombra:** Con esta opción, podremos aplicar sombras a las figuras seleccionadas. El cuadro de diálogo que se nos abre nos muestra las siguientes opciones:
  - **Estilo:** Establece el estilo de sombra para la forma seleccionada. Con Valor predeterminado de página, la forma adopta la configuración del estilo sombra de la página, que podemos ver en la opción *configurar página*.
  - **Color:** Determina el color de fondo de la sombra de la forma seleccionada. Seleccionaremos en la lista un nuevo color de fondo o haremos clic en *Más colores* para definir un color personalizado en el cuadro de diálogo *Colores*.
  - **Trama:** Determina la trama de la sombra. Seleccionaremos *Ninguna* para eliminar una sombra existente.
  - **Color de trama:** Determina el color de trama de la sombra. Seleccionaremos en la lista un nuevo color de fondo o haremos clic en *Más colores* para definir un color personalizado en el cuadro de diálogo *Colores*.
  - **Transparencia:** Moveremos el control deslizante para establecer, del 0 al 100 por ciento, la transparencia de la sombra.
  - **Vista previa:** Muestra una vista previa de las sombras seleccionadas.

- **Tamaño y posición:** Especifica la posición de las sombras con respecto a las formas. Para cambiar el desplazamiento de la sombra, escribiremos otra distancia en los cuadros *Desplaz. X* y *Desplaz. Y*. Podremos hacer clic en las flechas para comprobar visualmente los valores de desplazamiento. La opción *ampliación* especifica el tamaño de la sombra en relación con la forma.
- **Dirección:** Especifica el ángulo de una sombra oblicua.

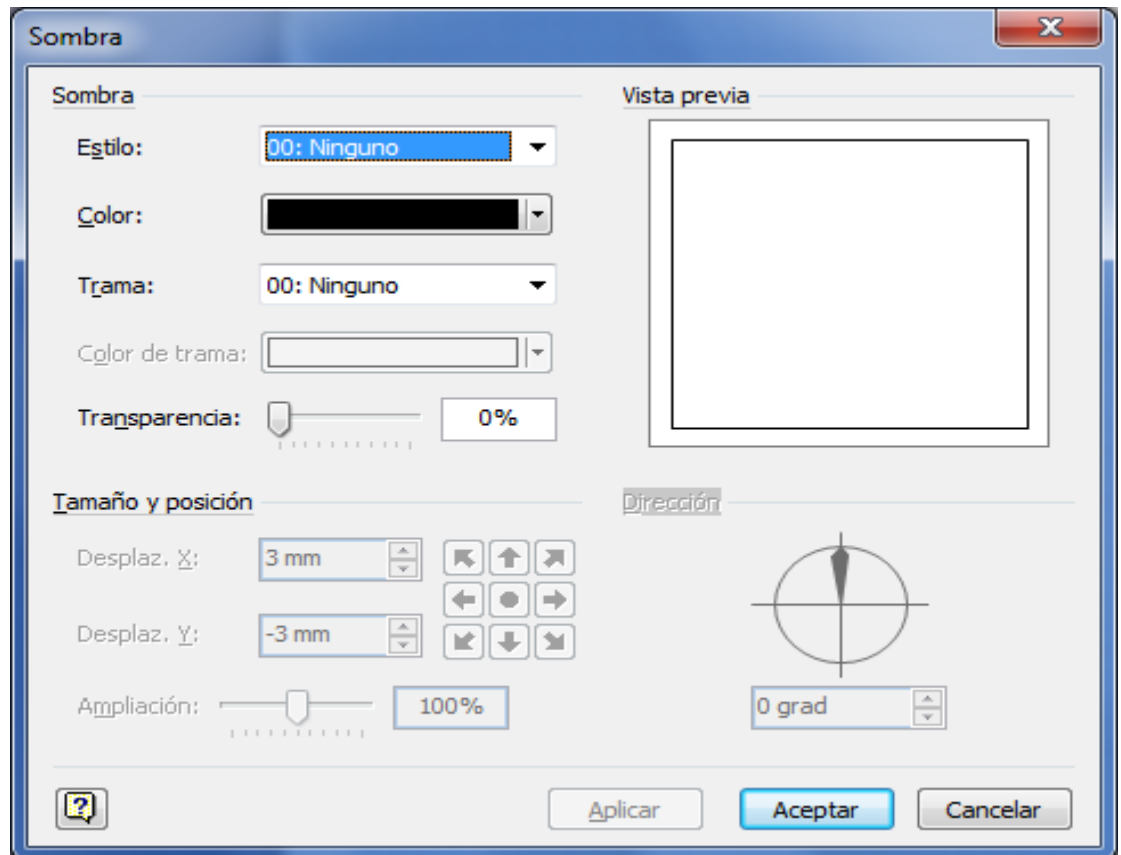


Figura 43: Vista de la opción Sombra

- **Exportar a base de datos:** Con esta opción, podremos exportar los datos de las figuras a una base de datos. El cuadro de diálogo que se nos abre nos muestra las siguientes opciones:
  - **Nombre:** Elegiremos un origen de datos si lo tenemos ya creado.
  - **Crear:** Para crear un nuevo origen de datos, haremos clic en Crear y seguiremos el asistente Crear nuevo origen de datos. Una vez finalizado, aparecerá el cuadro de diálogo Configuración de ODBC, desde el cual podremos seleccionar una base de datos existente o crear una nueva. La tabla de la base de datos se creará más adelante.



- **Examinar.** Si el origen de datos no se halla almacenado en la ubicación predeterminada, utilizaremos la opción examinar para buscarlo.
- **Calificador:** Seleccionaremos la base de datos que tenga la tabla que queramos vincular. Esta opción sólo está disponible para programas de bases de datos, como Microsoft SQL Server, donde, si nos conectamos a un origen de datos, tendremos acceso a varias bases de datos.
- **Propietario.** Podremos elegir Todos los usuarios o el nombre del creador de la base de datos. La selección de un creador concreto restringe el número de tablas que se muestran en la lista Tabla. Esta opción sólo está disponible para programas de bases de datos, como Microsoft SQL Server, que permiten especificar un creador de objeto.
- **Tabla:** Elegiremos la tabla a la que deseemos exportar los datos. Para crear una tabla nueva y exportar a ella los datos, escribiremos un nombre para la tabla nueva.
- **Campo:** Crea un campo clave llamado ShapeKey en la tabla de la base de datos de manera predeterminada. Podremos escribir otro nombre si queremos cambiarlo.
- **Tipo:** Especifica los valores para el campo clave.
  - ❖ Elegiremos ShapeID para utilizar un número que se asigna a una forma de acuerdo al orden en que ésta se creó en la página. Dos formas de la misma página de dibujo nunca tendrán el mismo ShapeID.
  - ❖ Elegiremos GUID para utilizar un número único de 128 bits terminado en cero que se asigna a cada forma cuando exporta sus datos. Dos formas del mismo documento nunca tendrán el mismo GUID.
- **Convertir el campo de clave en la clave principal de la tabla:** Si convertimos el campo de clave en un campo de clave principal, la velocidad de acceso al campo del programa de base de datos será mayor. No todos los orígenes de datos ODBC pueden marcar campos como campos de clave principal.
- **Celdas para exportar:** Enumera las celdas que se van a exportar.
  - ❖ **Celda:** El nombre de la celda que se va a exportar.
  - ❖ **Evaluar como:** Muestra cómo se evalúan los datos al copiarlos de la celda al campo de la base de datos.
  - ❖ **Campo:** El nombre del campo de la base de datos al que los datos de la forma se van a copiar.
  - ❖ **Tipo:** El tipo de datos para el campo de la base de datos. Los tipos de datos que puede usar dependen del programa de base de datos al que se desea vincularlos.
  - ❖ **Longitud:** Escribiremos el tamaño en bytes de un campo de longitud variable.
  - ❖ **Decimal:** Escribiremos el número de decimales que admite el campo. Esto se aplica exclusivamente a los campos numéricos que no sean enteros.
- **Agregar:** Haremos clic para agregar una nueva celda a la lista de celdas exportadas.

- **Modificar:** Haremos clic para modificar el vínculo entre la celda seleccionada y un campo de la base de datos.
- **Eliminar:** Haremos clic para quitar la celda seleccionada de la lista de celdas exportadas.

Los pasos a seguir para la exportación son los siguientes:

1. Seleccionaremos la tabla de base de datos que deseemos utilizar. Si su origen de datos no se halla almacenado en la ubicación predeterminada, haremos clic en Examinar para localizarlo.
  - Si utilizamos Microsoft Excel como origen de datos, podremos seleccionar un libro existente pero no podremos crear uno nuevo mediante el asistente.
  - Si seleccionamos un libro de Excel, debemos asegurarnos de desactivar la casilla de verificación “Sólo lectura” en el cuadro de diálogo Seleccionar libro.
2. Si la base de datos admite la utilización de propietarios, haremos clic en el nombre de uno en la lista propietarios para filtrar la lista de propietarios de la tabla. La tabla de la base de datos se crea con el nombre utilizado para iniciar sesión en ella.
3. Si la base de datos proporciona acceso a diferentes bases de datos, seleccionaremos la que desee utilizar en la lista calificador.
  - Si utilizamos Microsoft Access o Excel, el nombre y la ubicación de la base de datos a la que se exportarán los datos aparecerán aquí.
4. En la lista Tabla, haremos clic en la tabla de la base de datos que desea utilizar. Si queremos crear una nueva tabla, escribiremos un nombre.

Si queremos cambiar el formato de exportación de los datos, podremos hacerlo con los botones *agregar*, *modificar* o *eliminar*.

Debemos tener cerrada la base de datos a la hora de exportar los datos, ya que de lo contrario, no nos dejará y se producirá un error.

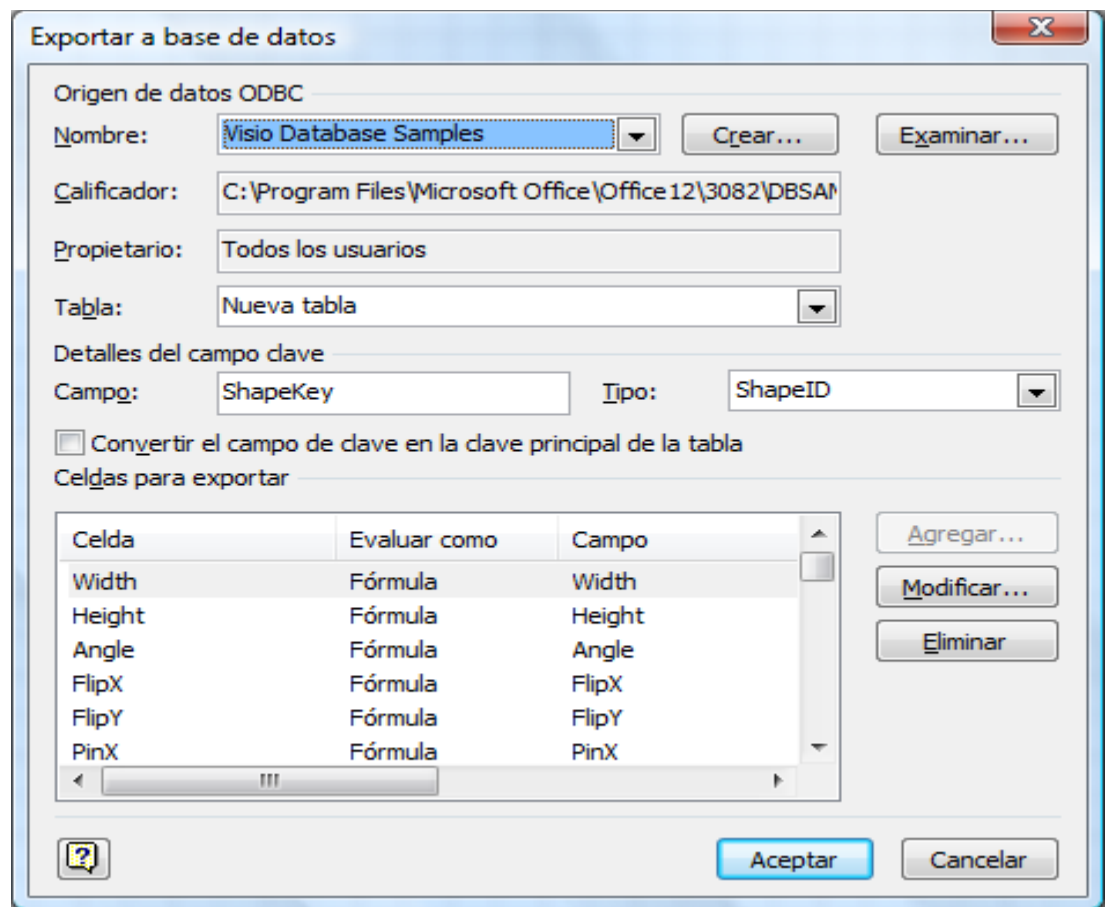


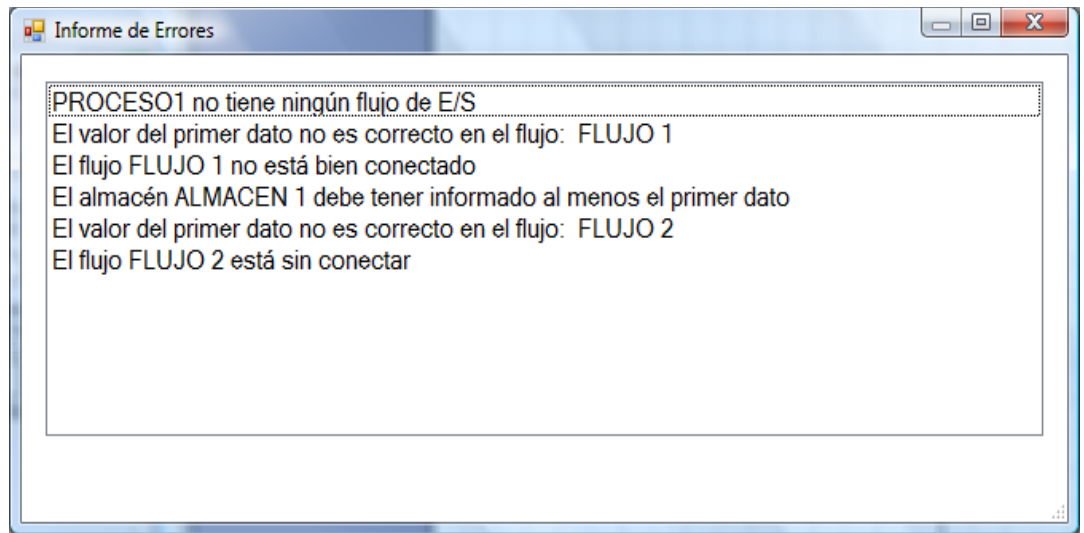
Figura 44: Vista de la opción Exportar a base de datos

➤ Opciones

- **Chequear:** Esta opción se utiliza para chequear el nivel actual y detectar cualquier error lógico, de diseño o metodológico. Después de realizar el chequeo, visualizaremos una lista con los errores que se produzcan o un mensaje indicando si ha sido correcto.

Se comprobará lo siguiente:

- ❖ Que no existan figuras sin nombrar.
- ❖ Que no existan procesos repetidos.
- ❖ Que no existan procesos sin flujos conectados y que como mínimo tengan un de entrada y uno de salida.
- ❖ Que los almacenes y las entidades externas no estén conectados entre sí.
- ❖ Que los datos de los flujos estén en los almacenes a los que están conectados.

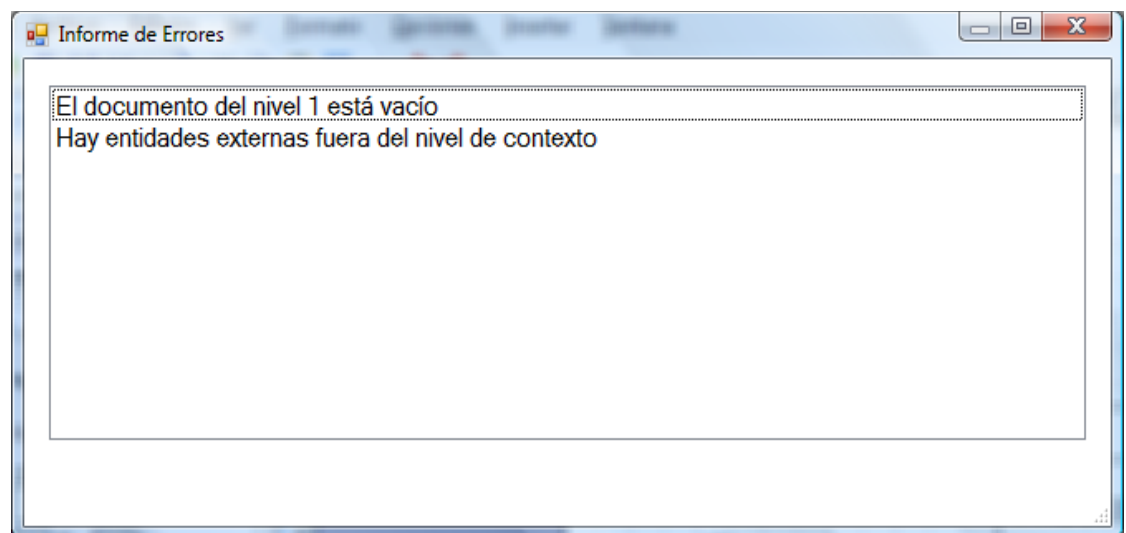


*Figura 45: Informe de la opción Chequear*

- **Balancear:** Con esta opción chequearemos el correcto balanceo de todo el diagrama. Después de realizar el chequeo se visualiza una lista con los errores que se han producido o un mensaje indicando si ha sido correcto.

Se comprobará lo siguiente:

- ❖ Que no existan figuras sin nombrar.
- ❖ Que las explosiones sean correctas.
- ❖ Que no existan procesos primitivos en el nivel de contexto.
- ❖ Que no existan entidades externas fuera del nivel de contexto.



*Figura 46: Informe de la opción Balancear*

➤ Insertar

- **Nueva página:** Cuando creamos un diagrama nuevo, siempre se abre con una sola página, pero con esta opción, podremos agregar las páginas que queramos a nuestro diagrama, que irán en función del número de niveles que tengamos. Cada vez que insertemos una página mediante esta opción, automáticamente se nos abrirá el cuadro de diálogo de la opción *Configurar página*, para hacer una configuración personalizada de la nueva página. Para más información, revisar esta opción en el manual.
- **Texto:** Con esta opción podremos insertar texto al diagrama que no dependa de ninguna figura. Después de pulsar esta opción, haremos clic en cualquier lugar de la página, o clic y arrastramos hasta que el bloque de texto tenga el tamaño que deseemos y después escribiremos el texto.

➤ Ventana

Con esta opción veremos todos los diagramas que tenemos abiertos en ese momento. Si pulsamos sobre alguno de ellos, se nos pondrá como el diagrama actual, es decir, le asignaremos el foco y los demás quedarán en un segundo plano.

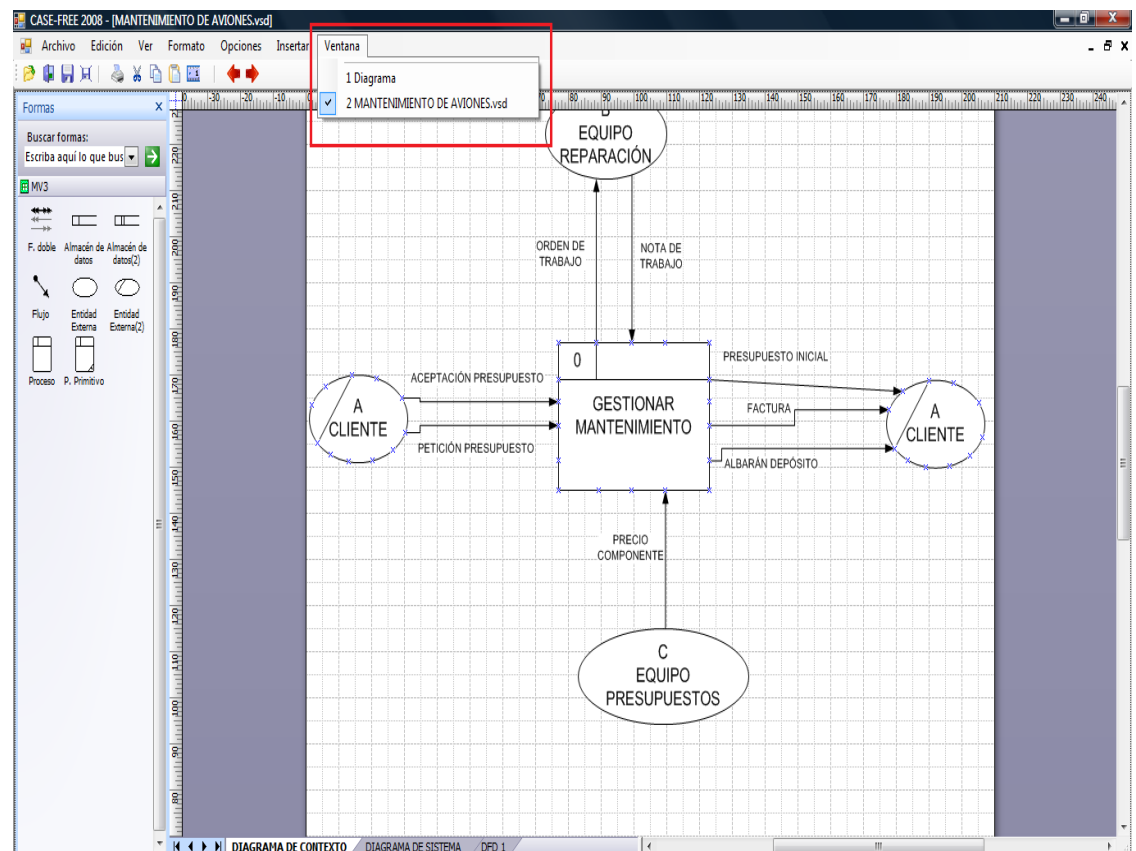


Figura 47: Vista de la opción Ventana

## 7.4. Opciones de ratón

Aparte de las opciones de menú explicadas en el apartado anterior, también disponemos de otras opciones que podemos ejecutar con el ratón. Algunas de estas son iguales, pero otras sólo estarán disponibles mediante el ratón. Para ello debemos seleccionar una figura del diagrama y hacer clic con el botón derecho. Las opciones que nos aparecen son:

- **Cortar:** Misma opción que en la barra del menú edición. Véase apartado anterior.
- **Copiar:** Misma opción que en la barra del menú edición. Véase apartado anterior.
- **Pegar:** Misma opción que en la barra del menú edición. Véase apartado anterior.
- **Formato:**
  - **Texto:** Misma opción que en la barra del menú formato. Véase apartado anterior.
  - **Línea:** Misma opción que en la barra del menú formato. Véase apartado anterior.
  - **Relleno:** Misma opción que en la barra del menú formato. Véase apartado anterior.
  - **Permitir temas:** No aplica.

- **Protección:** Especifica las características de la forma seleccionada que deseemos bloquear:
  - **Ancho:** Bloquea el ancho de la forma seleccionada para impedir que se cambie su tamaño. Para bloquear una forma contra el giro, deberemos bloquear su ancho.
  - **Alto:** Bloquea el alto de la forma seleccionada para impedir que se cambie su tamaño.
  - **Relación de aspecto:** Bloquea la relación entre el alto y el ancho de una forma para que se mantengan sus proporciones al modificar su tamaño.
  - **Posición X:** Bloquea la posición x (horizontal) de una forma para que no se pueda mover.
  - **Posición Y:** Bloquea la posición y (vertical) de una forma para que no se pueda mover.
  - **Giro:** Bloquea una forma para que no se pueda girar con la herramienta Giro. No obstante, podremos girar una forma arrastrando sus extremos. Para bloquear una forma contra giro, debemos bloquear su ancho.
  - **Punto inicial:** Bloquea el punto inicial de una forma en una ubicación específica para que no se pueda mover.
  - **Punto final:** Bloquea el punto final de una forma en una ubicación específica para que no se pueda mover.
  - **Texto:** Bloquea el texto de una forma para que no se pueda modificar.
  - **Formato:** Bloquea el formato (de línea, texto, relleno, sombra, redondeo de esquinas y transparencia) de una forma para que no se pueda modificar.
  - **Contra selección:** No aplica.
  - **Contra eliminación:** Bloquea la forma para que no se pueda eliminar.
  - **Contra formato en grupo:** Bloquea el formato (color, relleno, línea y sombra) en subformas para que no cambien si se da formato a la forma agrupada. Sigue pudiendo dar formato a una subforma seleccionándola.
  - **Contra colores del tema:** No aplica.
  - **Contra efectos del tema:** No aplica.
  - **Todo:** Bloquea todas las opciones de *Proteger*.
  - **Nada:** Desbloquea todas las opciones de *Proteger*.

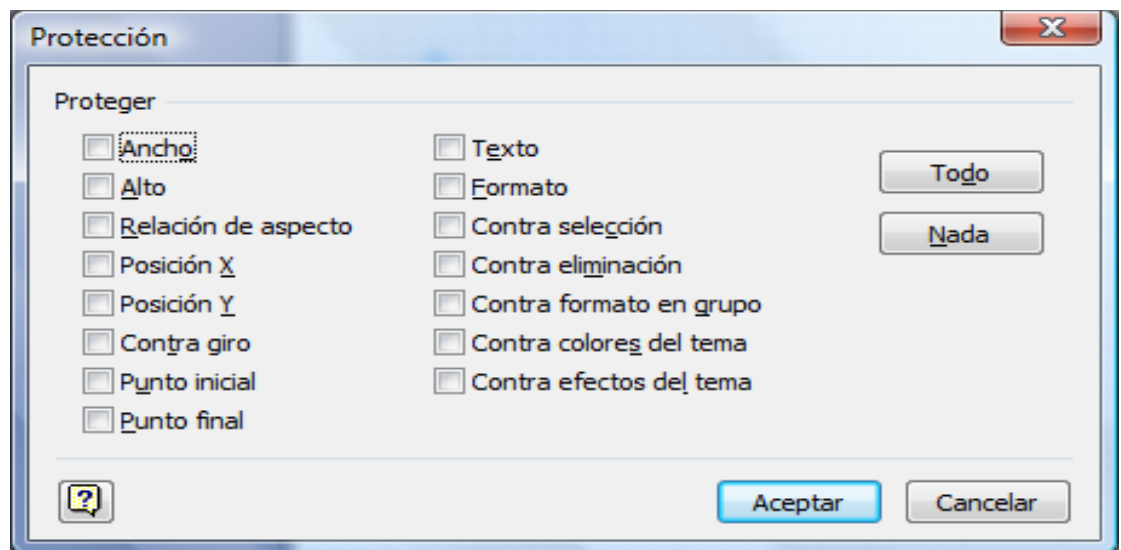


Figura 48: Vista de la opción Protección

- **Comportamiento:** Especifica el comportamiento que puede tener la figura seleccionada. El cuadro de diálogo está dividido en tres pestañas:

- **Comportamiento:**

- ❖ **Estilo de interacción:** Especifica si una forma se comporta como una línea (forma unidimensional) o como un cuadro (forma bidimensional).
- ❖ **Resaltar la selección:** Especifica si se muestran los controladores, los controladores de formas y los cuadros de alineación al seleccionar una forma.
- ❖ **Comportamiento al cambiar el tamaño:** Determina cómo cambia el tamaño de una forma o de otro objeto de un grupo al cambiar el tamaño del grupo.
- ❖ **División de conectores:** No aplica.
- ❖ **Varios:** Si seleccionamos *Forma no imprimible*, la forma aparecerá en la pantalla, pero no en la copia impresa.

Si seleccionamos *Agregar formas a un grupo al colocarlas*, al arrastrar una forma a un grupo existente pasará a formar parte de él. También debemos seleccionar la opción *Aceptar formas colocadas*, en *Comportamiento del grupo*.

- ❖ **Comportamiento del grupo:** Si la opción *Ajustar a las formas pertenecientes* está seleccionada, podemos ajustar y pegar la forma a las formas de un grupo.

Si está seleccionada la opción *Modificar el texto del grupo*, podemos modificar el texto de un grupo.

Cuando la opción *Aceptar formas colocadas* está seleccionada, indica que el grupo aceptará las formas que se arrastren a él y las incorporará al grupo. También debemos seleccionar la opción *Agregar formas a un grupo al colocarlas*, en varios, para cada una de las formas.

- ❖ **Selección:** Especifica qué sucede al seleccionar un grupo.



- **Sólo grupo:** Especifica que al hacer clic en un grupo, sólo se selecciona el grupo, y no las formas individuales que hay en él.
- **Primero grupo:** Especifica que al hacer clic en un grupo, se selecciona primero el grupo. Si hace clic de nuevo, puede seleccionar una forma específica.
- **Primero miembros:** Especifica que al hacer clic en un grupo, se selecciona la forma concreta a la que se apunta. Para seleccionar el grupo tiene que hacer clic en el cuadro delimitador que hay alrededor de dicho grupo.
- ❖ **Datos del grupo:** Establece el orden de presentación del grupo para el texto o las formas que hayamos creado.
  - **Ocultar:** Oculta el texto o las formas del grupo que hayamos creado.
  - **Detrás de los miembros:** Coloca los componentes del grupo detrás de las formas del grupo. Por ejemplo, si el grupo contiene texto, éste se mostrará detrás de las formas agrupadas.
  - **Delante de los miembros:** Coloca los componentes del grupo delante de las formas del grupo. Por ejemplo, si el grupo contiene texto, éste se mostrará delante de las formas agrupadas.
- **Doble clic:**
  - ❖ **Ejecutar la acción predeterminada:** Especifica que al hacer doble clic en la forma se activa la acción predeterminada para esa forma, como abrir el bloque de texto de manera que pueda modificar el texto.
  - ❖ **No ejecutar ninguna acción:** Especifica que al hacer doble clic en la forma no se realiza ninguna acción.
  - ❖ **Modificar el texto de la forma:** Especifica que al hacer doble clic en la forma se abre el bloque de texto de la forma.
  - ❖ **Abrir grupo en ventana nueva:** No aplica.
  - ❖ **Abrir la forma en ShapeSheet:** No aplica.
  - ❖ **Personalizado:** No aplica.
  - ❖ **Mostrar la Ayuda:** Especifica que al hacer doble clic en la forma se muestra un tema de ayuda para la forma. Este tema puede ser diferente del que aparece al hacer clic con el botón secundario en la forma y seleccionar *Ayuda*. La sintaxis que debe utilizar es la siguiente:  
**FILENAME!keyword or FILENAME!#Number**
    - **Filename:** es el archivo de Ayuda de Windows (puede ser un archivo .hlp o .chm).
    - **Keyword:** es el término de índice asociado al tema de Ayuda.
    - **Number:** es un identificador numérico al cual se hace referencia en la sección MAP del archivo de proyecto de Ayuda.

- ❖ **Verbo OLE:** Especifica que al hacer doble clic en un objeto OLE (OLE: protocolo de Windows que permite incrustar un objeto creado en una aplicación de Windows en un documento creado en otra aplicación de Windows diferente, o bien vincular un objeto a un archivo que contiene el objeto original.) se activan las opciones de OLE, como abrir el programa de origen para que pueda modificar el objeto.
- ❖ **Ejecutar macro:** Especifica que al hacer doble clic en la forma se ejecuta la macro que seleccione de la lista.
- ❖ **Ir a la página:** Especifica que al hacer doble clic en la forma se muestra una página de dibujo específica.
- ❖ **Abrir en ventana nueva:** No aplica.
- **Colocación:** Especifica el comportamiento de las formas bidimensionales durante la disposición automática.
  - ❖ **Comportamiento al colocar:** Determina el modo de interacción de una forma 2D con un flujo y si se incluye en una disposición automática.
    - **Que decida Visio:** Permite que Microsoft Visio determine si convertirá la forma en colocable, según el tipo de flujo que pegue a la forma.
    - **Disponer y rodear figura:** Convierte una forma bidimensional en colocable en todos los casos. Los flujos siempre la detectan, y siempre está incluida en una disposición automática.
    - **No disponer ni rodear figura:** Hace que una forma bidimensional nunca sea colocable. Los flujos no pueden detectarla, y no está incluida en una disposición automática.
  - ❖ **Colocación:**
    - **No mover al colocar:** Especifica que no se debe mover la forma durante la disposición automática.
    - **Permitir que otras formas se coloquen encima:** Especifica que se pueden colocar otras formas encima de la seleccionada durante la disposición automática.
  - ❖ **Mover formas al colocar:**
    - **Mover las otras formas al colocar:** Especifica si la forma aparta otras formas de su camino al cambiar de posición en la página
    - **Apartar según página:** Significa que la forma se comportará según la configuración de la página.
    - **No apartar ninguna forma:** Significa que la forma nunca aparta otras formas.
    - **Apartar todas las formas:** Significa que la forma siempre aparta otras formas.

- **No permitir que otras formas muevan ésta al colocarse:** Especifica que la forma seleccionada no se moverá cuando se arrastren otras formas a la página.

❖ **Interacción con conectores:**

- **Enrutar horizontalmente:** Especifica que un flujo puede atravesar una forma bidimensional horizontalmente (una línea a través de su centro).
- **Enrutar verticalmente:** Especifica que un flujo puede atravesar una forma bidimensional verticalmente (una línea a través de su centro).

Si la figura seleccionada es un flujo, aparecerá una pestaña nueva llamada **Conector**, que especifica el comportamiento de un flujo:

- ❖ **Estilo:** Establece el estilo de enrutamiento del flujo seleccionado.
- ❖ **Dirección:** En algunos estilos de enrutamiento de flujos podemos establecer la dirección o el flujo del estilo de enrutamiento.
- ❖ **Volver a enrutar:** Especifica cuándo se vuelve a enrutar el flujo seleccionado.
  - **Libremente:** El flujo se vuelve a enrutar siempre que es necesario.
  - **Según sea necesario:** El flujo se vuelve a enrutar cuando se coloca en su ruta una forma colocable o cuando un segmento del flujo atraviesa una forma a la que está conectado.
  - **En el cruce:** El flujo se vuelve a enrutar cuando uno de sus segmentos cruza una forma a la que está conectado. Cuando cambia manualmente la ruta de un flujo, se asigna este valor al conector.
  - **Nunca:** El flujo nunca se vuelve a enrutar, ni siquiera cuando uno de sus segmentos cruza una forma a la que está conectado.
- ❖ **Apariencia:** Especifica si los flujos son rectos o curvados.
- ❖ **Agregar:** Especifica cuándo se agregarán saltos de línea al flujo seleccionado. Esta opción interactúa con la opción Agregar saltos de línea a, en la ficha Disposición y enrutamiento del cuadro de diálogo Configurar página.
  - **Según página:** Agrega un salto de línea al flujo seleccionado según lo especifique la configuración de la página.
  - **Nunca:** Impide los saltos de línea en el flujo seleccionado, aunque normalmente deba incluirlos.
  - **Siempre:** Agrega un salto de línea al flujo seleccionado, aunque normalmente no deba incluirlos.
  - **Siempre a otra:** Impide los saltos de línea en el flujo seleccionado y se agregan al conector con el que se cruza.
  - **A ninguna de las dos:** Impide que se incluyan saltos de línea en el flujo seleccionado y en el flujo con el que se cruza.
- ❖ **Estilo:** Especifica la apariencia de los saltos de línea del flujo seleccionado.

❖ **Vista previa:** Nos ofrece una vista previa de las opciones seleccionadas.

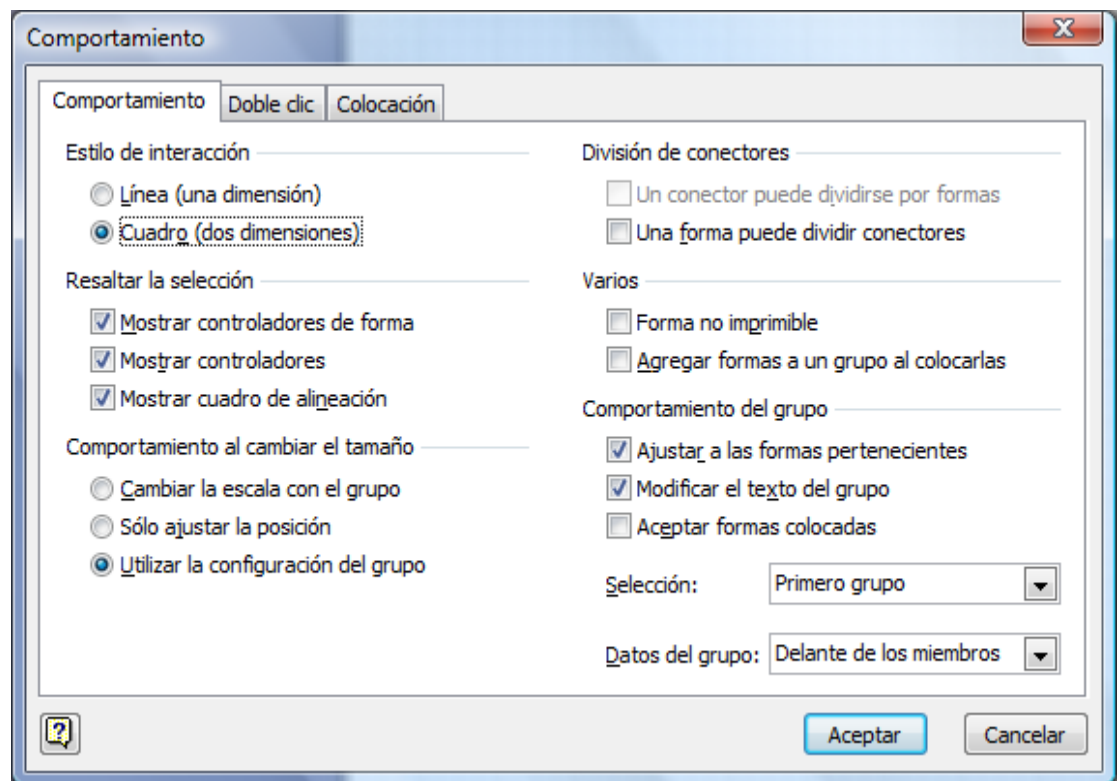


Figura 49: Vista de la opción Comportamiento

- **Capa:** Asigna la forma seleccionada a una o más capas (capa: categoría con nombre de formas. Para organizar las formas del dibujo debemos asignarlas a capas. Podemos ver, modificar, imprimir o bloquear formas de manera selectiva en las capas.) o quitar una asignación de capa de una forma.
- **En las capas:** Resalta las capas a las que la forma seleccionada está asignada. Si deseamos cambiar la asignación de una capa, haremos clic en el nombre de una capa distinta. Para asignar la forma a otras capas, presionaremos *Ctrl* y haremos clic en sus nombres.
- **Mantener las capas de los miembros del grupo:** Especifica que las formas de los miembros del grupo conservan las asignaciones de capa actuales. Esta opción se convierte en activa al asignar el grupo a una capa. La selección de esta opción sólo tendrá efecto cuando la forma seleccionada actual sea un grupo.
- **Todas:** Seleccionamos todas las capas disponibles en la lista.
- **Ninguna:** Quitamos todas las selecciones de capas activas en la lista.
- **Nueva:** Abre el cuadro de diálogo Nueva capa, en el que podemos escribir un nombre para una capa nueva.

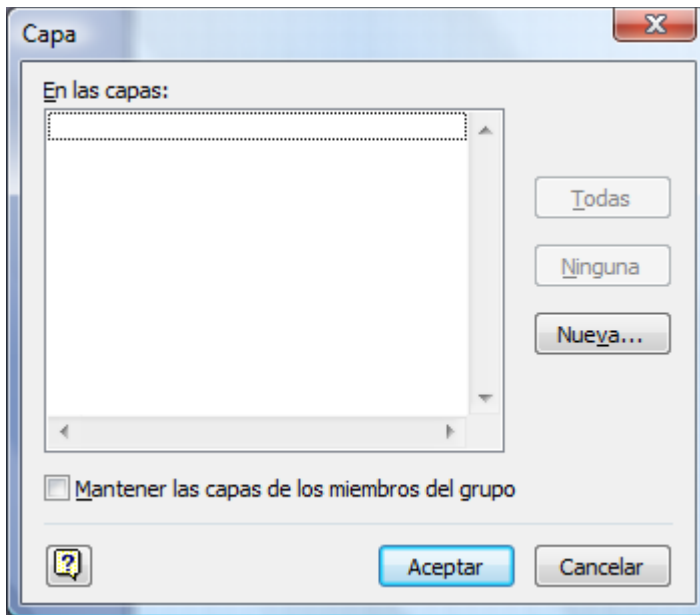


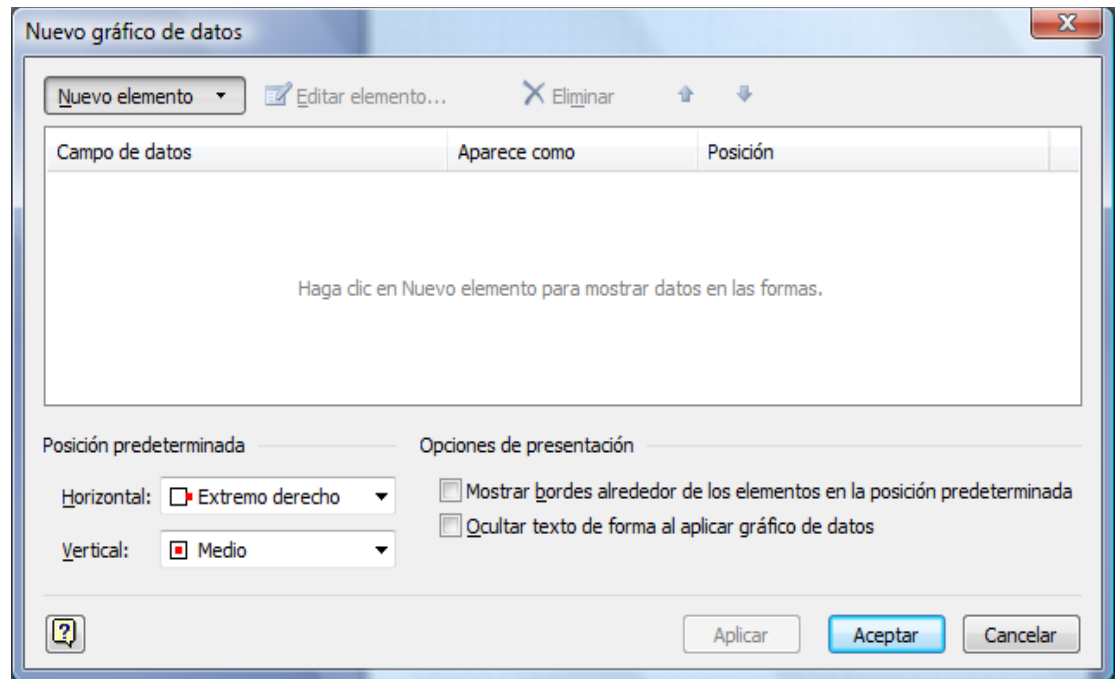
Figura 50: Vista de la opción Capa

○ **Datos:**

- **Datos de formas:** Véase opción de menú propiedades en el apartado anterior.
- **Editar gráfico de datos:** Los gráficos de datos son un conjunto de opciones que se pueden aplicar a las figuras para mostrar los datos que contienen. El gráfico de datos presenta los datos como una combinación de elementos textuales y visuales, por ejemplo, indicadores y barras de progreso. Tenemos cuatro tipos de elementos para el gráfico, que son: texto, barras de datos, iconos y códigos de color. Para los tres primeros pasos a seguir son los siguientes:
  5. En el cuadro de diálogo Editar gráficos de datos, haremos clic en *Nuevo elemento* y después en *Texto*.
  6. En el cuadro de diálogo *Nuevo texto*, haremos clic en *Campo de datos* y seleccione un elemento de la lista.
  7. Hacer clic en *Llamada* y seleccionar un estilo del conjunto de iconos en la lista.
  8. Seleccionar la posición de la llamada en una de las formas siguientes:
    - En *Posición de la llamada*, activar la casilla de verificación *Usar posición predeterminada*.
    - Desactivar la casilla de verificación *Usar posición predeterminada* y después hacer clic en los cuadros de texto *Horizontal* y *Vertical* y seleccionar las posiciones en las listas.
  9. En *Detalles*, seleccionar las opciones para elegir el aspecto de la llamada o del encabezado.

Para la opción color por valor haremos lo siguiente:

1. En el cuadro de diálogo *Editar gráficos de datos*, haremos clic en *Nuevo elemento* y después haga clic en *Color por valor*.
  2. En el cuadro de diálogo *Nuevo color por valor*, en *Presentación*, seleccionar un elemento de la lista *Campo de datos*.
  3. En *Método de colorear*, seguir uno de estos procedimientos:
    - Seleccionar *Cada color representa un valor único* para aplicar el mismo color a todas las formas que tengan el mismo valor.  
La lista *Asignaciones de color* muestra todos los valores disponibles y asigna un color de relleno y un color de texto a cada valor. Para seleccionar un color de texto o de relleno diferente, haremos clic en la flecha *Color de relleno* o *Color del texto*.
    - Seleccionar *Cada color representa un intervalo de valores* para usar un intervalo de colores, de brillo a mate, para indicar un intervalo de valores.  
La lista *Asignaciones de color* muestra una lista de posibles intervalos de valores y asigna un color de relleno y un color de texto a cada intervalo. Especificar valores diferentes en los cuadros Valor 1 y Valor 2 para crear intervalos personalizados. Para seleccionar distintos colores de texto y de relleno para cada valor, podemos hacer clic en la flecha Color de relleno o Color del texto. Sin embargo, para conservar el efecto brillo a mate de los conjuntos de color, se recomienda elegir otro color para sólo el primer elemento en la lista Asignaciones de color.
  4. Hacer clic primero en *Aceptar* en el cuadro de diálogo *Nuevo color por valor* y después en el cuadro de diálogo *Editar gráfico de datos* para aplicar la configuración a las formas del dibujo.
- **Quitar gráfico de datos:** Esta opción eliminará los gráficos de datos.



*Figura 51: Vista de la opción Gráfico de datos*

○ **Forma:**

- **Agrupar:** Esta opción agrupará las figuras que seleccionemos para moverlas a la vez o bien aplicarles el mismo formato a todas. A efectos, es como si todas las formas agrupadas fueran una.
- **Desagrupar:** Esta opción desagrupa las formas que teníamos agrupadas.
- **Traer al frente:** Si tenemos varias formas unas encima de otras, con esta opción, la forma seleccionada se pondrá por encima de todas las demás.
- **Enviar al fondo:** Si tenemos varias formas unas encima de otras, con esta opción, la forma seleccionada se pondrá por debajo de todas las demás.
- **Traer hacia adelante:** Si tenemos varias formas unas encima de otras, con esta opción, la forma seleccionada se pondrá por encima justo de la que tenía delante.
- **Enviar hacia atrás:** Si tenemos varias formas unas encima de otras, con esta opción, la forma seleccionada se pondrá por debajo justo de la que tenía detrás.
- **Girar texto:** Con esta opción giraremos el texto de la figura que tengamos seleccionada o de bloques de texto independiente.
- **Buscar formas similares:** Esta opción solo está disponible si la selección es un flujo. Buscará formas similares entre las plantillas que dispongamos.

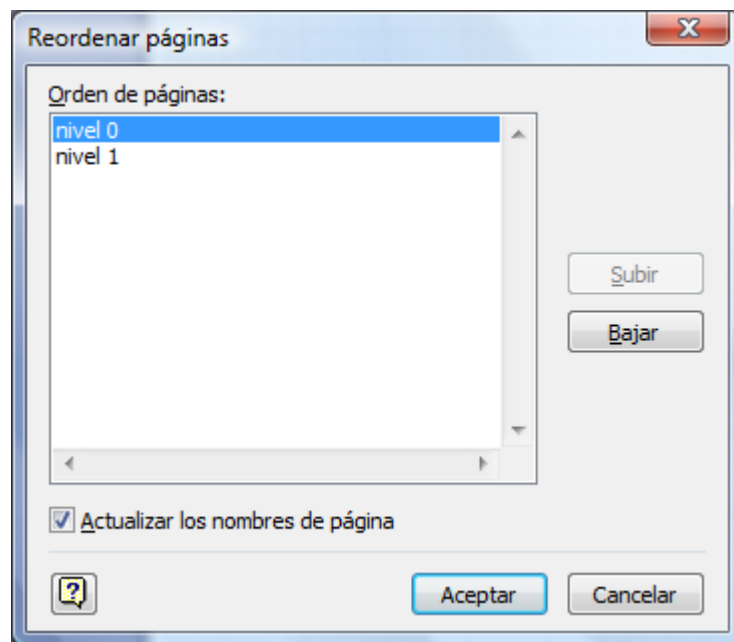
Si la figura que seleccionamos es un flujo, dispondremos de las siguientes opciones:

- **Conector de ángulo recto:** Podremos moldear el flujo con forma de ángulo recto.
- **Conector recto:** El flujo siempre será una línea recta.

- **Conector curvado:** Podremos moldear el flujo con curvas.
- **Restablecer conector:** El flujo volverá a la posición original.
- **Ayuda:** Se abrirá un cuadro de diálogo de ayuda.

En la parte inferior de la aplicación, tenemos el nombre de la página, si hacemos clic con el botón secundario del ratón tendremos las siguientes opciones:

- **Insertar página:** Véase opción insertar página del apartado anterior.
- **Eliminar página:** Véase opción eliminar páginas del apartado anterior.
- **Cambiar nombre de página:** Podremos renombrar la página.
- **Reordenar página:** Con esta opción podremos mediante una caja de diálogo, cambiar el orden de las páginas.
  - ❖ **Orden de las páginas:** Enumera todas las páginas por su nombre en el orden en que se almacenaron en el archivo de dibujo. Seleccionaremos una página para moverla.
  - ❖ **Actualizar los nombres de página:** Activaremos esta casilla de verificación para volver a numerar las páginas cuando las hayamos reordenado. (Sólo se aplica a las páginas que utilizan nombres predeterminados, como Página-1.)
  - ❖ **Subir:** Mueve la página seleccionada hacia el principio del archivo.
  - ❖ **Bajar:** Mueve la página seleccionada hacia el final del archivo.



*Figura 52: Vista de la opción Reordenar páginas*



## 7.5. Explosionar procesos

Para explosionar un proceso, debemos seleccionarlo primero, y luego con el botón secundario del ratón, entraremos en el menú Formato→ Comportamiento, y en la pestaña *doble clic*, marcaremos la opción *Ir a la página*, en ella seleccionaremos el nivel al cual queremos explosionar ese proceso. Una vez hecho esto, cada vez que hagamos doble clic en ese proceso, iremos directamente al nivel donde lo hemos explosionado. Debemos crear el nivel al que vamos a explosionar antes de realizar la explosión y revisar este paso si cambiamos posteriormente el nombre de la página.

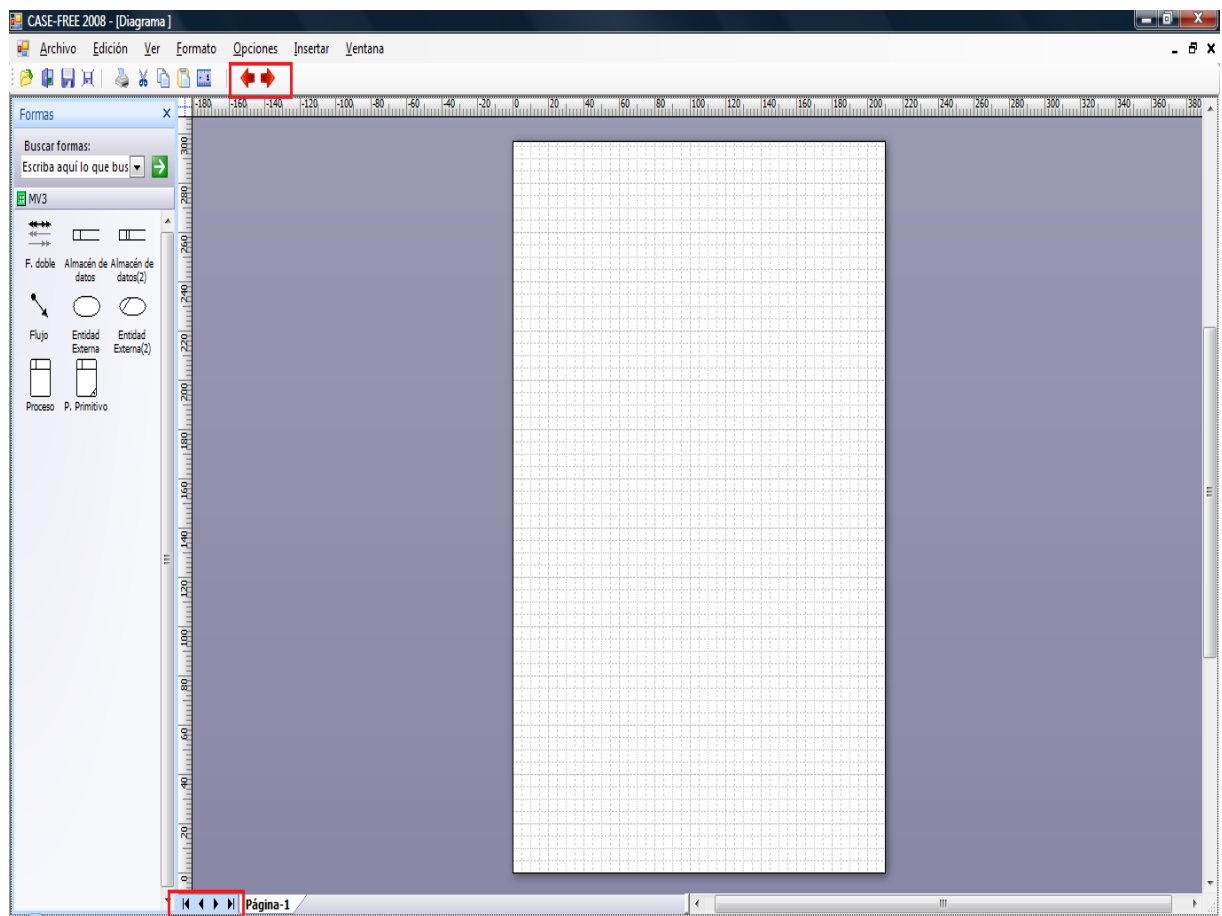
## 7.6. Nombrar procesos

Para nombrar una figura del diagrama, basta con hacer doble clic sobre ella y escribir un nombre o bien seleccionarla y pulsar la tecla *F2*, pero a los procesos además hay que darles una numeración, para ellos hay que realizar el siguiente paso:


- Con la opción de *Insertar*→ *Texto*, escribiremos el nombre y el número correspondiente en la parte que queramos del proceso, o bien lo podemos hacer fuera de él, y luego moverlo al lugar que queramos.

## 7.7. Otras utilidades

Podemos observar también que disponemos de una barra de herramientas con algunas de las opciones que hemos explicado anteriormente, con lo que tendremos un acceso más rápido. Estas opciones son *abrir*, *salir*, *guardar*, *guardar como*, *imprimir*, *cortar*, *copiar*, *pegar* y *regla*. Además, en esta barra tenemos una opción que no aparecía en ningún menú, que son las flechas de *avanzar* y *retroceder*, que servirán para movernos por los distintos niveles del diagrama de uno en uno. Esta navegación también la podremos hacer desde la parte inferior del diagrama, donde podremos avanzar o retroceder de nivel, ir al último o al primero, o bien acceder directamente al que queramos.



*Figura 53: Vista de la opción avance y retroceso*

En algunas opciones, podremos observar el siguiente símbolo , si pulsamos sobre él, se nos abrirá la ayuda donde podremos encontrar toda la información detallada en los puntos anteriores, y además, en algunos casos, más ampliada

## 8. Diseño del sistema

Como se ha comentado anteriormente en este documento, C# es un lenguaje de programación orientado a objetos que utiliza clases y estructuras.

Para la creación de nuestra aplicación, hemos creado seis clases, cuatro de ellas son formularios, otra está compuesta por dos estructuras y una interface, y la última una clase estática.

Vemos en el siguiente punto el diagrama de clases y a continuación una explicación de cómo funciona el sistema.

### 8.1. Diagrama de clases

Un diagrama de clases representa las clases que serán utilizadas dentro del sistema y las relaciones que existen entre ellas.

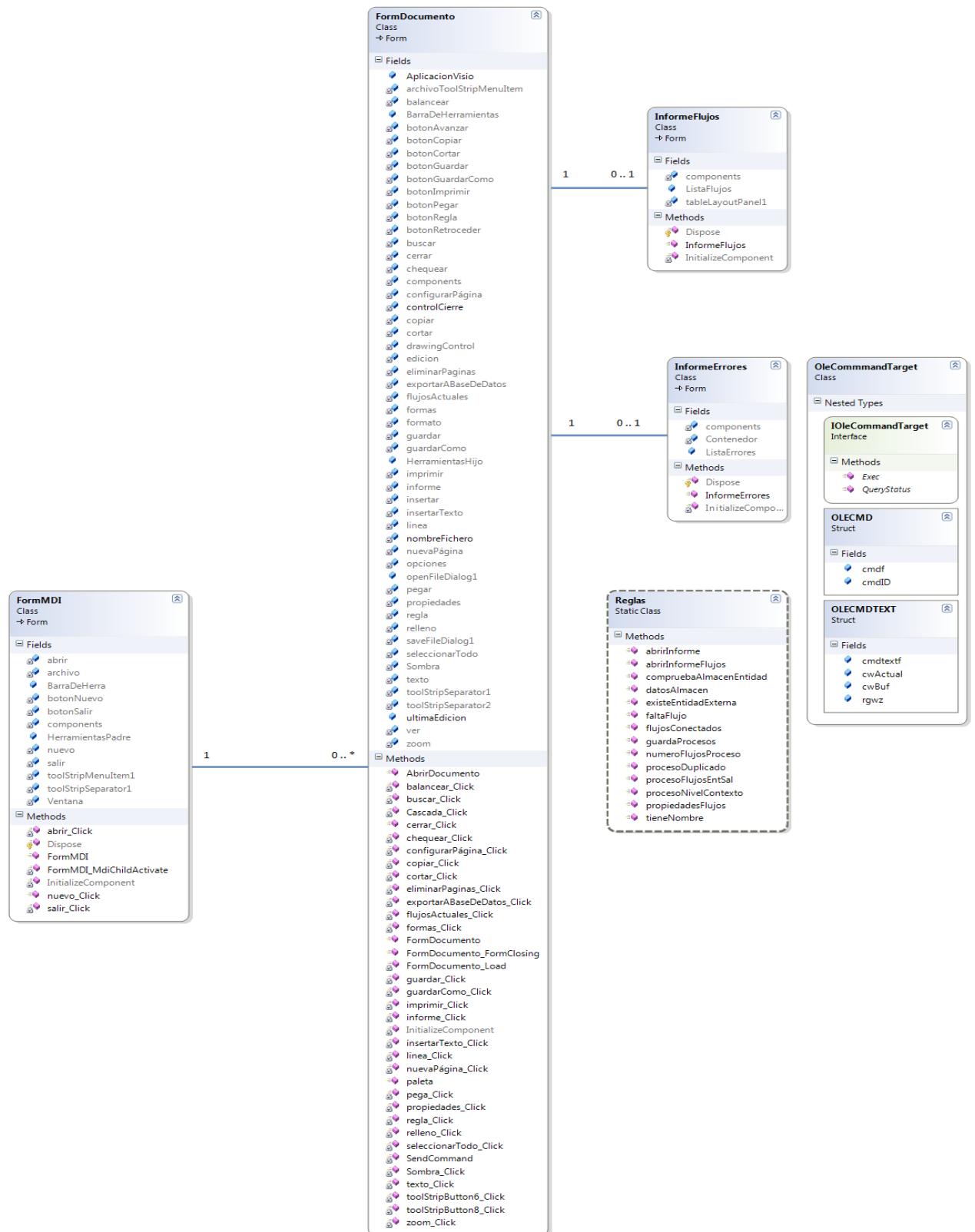


Figura 54: Diagrama de clases

## 8.2. Detalle de las clases

En este apartado daremos una descripción de cada clase y como interactúa dentro del sistema:

- **FormMDI:** La clase FormMDI, es una clase de tipo Form, es decir, un formulario. Además, es un tipo especial de formulario, es el formulario de interfaz de múltiples documentos (MDI), porque puede contener otros formularios denominados formularios MDI secundarios. También es conocido como formulario “Padre” o “Parent”, porque puede contener formularios llamados “Hijos” o “Child”, que son los formularios MDI secundarios que nombrábamos anteriormente.

Al abrir la aplicación, lo que estamos haciendo es instanciar esta clase, que corresponde a la primera interfaz que vemos.

- **FormDocumento:** También es una clase de tipo Form. Esta clase es la que denominábamos anteriormente como formularios MDI secundarios, es decir, los formularios “Hijos”. Cada vez que ejecutemos las opciones Nuevo o Abrir de la clase FormMDI, estaremos instanciando esta clase, y se abrirá un documento ya con todas las opciones de la aplicación. Su interfaz quedará contenida dentro de la interfaz de la clase FormMDI y se fusionarán sus barras de herramientas. Podremos instanciar esta clase, tantas veces como queramos, que corresponderán a todos los diagramas que tengamos abiertos en ese momento.

Como se puede apreciar en el diagrama de clases, esta clase es la que más atributos y métodos tiene de nuestro sistema, ya que en ella se agrupa casi toda la funcionalidad del aplicativo.

- **InformeFlujos:** También es una clase de tipo Form. Esta clase será instanciada cada vez que ejecutemos la opción del menú *Flujos Actuales*, y nos mostrará un informe con los flujos que hay en cada nivel del diagrama. La clase está compuesta por un ListBox, que es un contenedor donde vamos a ir almacenando la información para posteriormente mostrarla.
- **InformeErrores:** También es una clase de tipo Form. Esta clase será instanciada cada vez que ejecutemos las opciones del menú *Chequear* o *Balancear*, y nos mostrará un informe con los errores si los hubiera, tanto de diseño si se trata de la opción *Chequear*, como de balanceo si se trata de la opción *Balancear*. Esta clase también está compuesta por un ListBox, que es donde mostraremos el informe.

- **Reglas:** Esta clase es de tipo estático, por lo que no puede ser instanciada. Contiene todas las comprobaciones que se hacen tanto en la opción de menú *Balancear* como en la de *Chequear*, ya que muchas de estas comprobaciones son comunes en ambas opciones.
- **OleCommandTarget:** Esta clase está compuesta por una Interface y dos estructuras. La finalidad de esta clase es poder lanzar la funcionalidad de Visio. Esto lo haremos mediante un método intermedio, el cuál llamará a esta clase y así se lanzará la funcionalidad.

## 9. Conclusiones y líneas futuras

### 9.1. Conclusiones

Como principal conclusión, se destaca el hecho de que el presente proyecto haya satisfecho los objetivos fijados al inicio del mismo, construyendo una herramienta CASE bastante completa, donde se ha puesto especial cuidado en el diseño de una interfaz de usuario lo más fluida, intuitiva y atractiva posible.

Más específicamente, se ha conseguido materializar con éxito cada uno de los requisitos recogidos en la sección 4.1 (Requisitos de Usuario), y además con un aceptable rendimiento.

Con la realización de este proyecto he podido aprender más en profundidad el funcionamiento de las herramientas CASE, las cuales están cada día más extendidas por el mundo, y de las metodologías del desarrollo del software, en especial Métrica v3.

A pesar de parecer una aplicación aparentemente sencilla, lo que más ha costado de la programación es conseguir documentación y entenderla bien, de cómo integrar Microsoft Visio en nuestro entorno de desarrollo Visual Studio 2008. Apenas hay ejemplos en la documentación encontrada, y no existe ninguno donde se haya construido una aplicación a partir de esta integración. A pesar de esto, pienso que el poder integrar el paquete Microsoft Office dentro de Visual Studio, nos abre infinitud de posibilidades para construir multitud de aplicaciones.

Al empezar el proyecto, valoré cual era la mejor opción en cuanto a lenguajes de programación se refiere, y mis dos opciones siempre fueron C# y Java, dos lenguajes muy potentes, con un grado de similitud bastante grande y preparados para el desarrollo de aplicaciones gráficas. Del primero no sabía nada, del segundo si tenía conocimientos, pero al final me decanté por C#, por diversas razones. Lo primero, que me permitiría aprender un lenguaje nuevo de programación, el cual está creciendo y cada vez siendo más utilizado por las empresas, y lo segundo, que al ser de Microsoft me permitiría tener interoperabilidad con el paquete de Microsoft Office.

No obstante, considero que el sistema planteado ofrece aún múltiples posibilidades de ampliación por abordar en el futuro, algunas de las cuales se citarán en la siguiente sección, y es una de las razones por las que se ofrece este producto como un producto de software libre, para que cualquier usuario tenga libre acceso al código fuente y pueda modificarlo para potenciar y ampliar las posibilidades de la herramienta. Otra de las razones por la que es software libre, es para

que se puedan distribuir copias libremente entre los usuarios y todo el mundo que lo necesite pueda beneficiarse de sus ventajas.

Para concluir, quisiera destacar que a lo largo del desarrollo del presente proyecto, enmarcado en el ámbito de las nuevas tecnologías, el estudio y búsqueda continua de información que mejor se adaptara al cumplimiento de los objetivos marcados, ha propiciado un necesario aprendizaje de diversas herramientas, así como de los lenguajes de programación necesarios, lo cual ha supuesto, desde un punto de vista ya personal, un valor añadido de suma importancia sobre la formación recibida durante el transcurso de la titulación.

## **9.2. Líneas futuras**

Tras finalizar el presente proyecto software y valorar el alcance del mismo, se plantean las siguientes líneas futuras de ampliación:

- Actualización de la interfaz gráfica para optimizar interactividad y efectos visuales, y posibilitar la incorporación de futuras funcionalidades multimedia.
- Ampliación para soportar más metodologías de desarrollo.
- Ampliación para soportar más fases del ciclo de vida.
- Hacer de la herramienta una herramienta multi-idioma, incorporando los principales idiomas del mundo.



## 10. Glosario de términos y definiciones

### A

**ADA:** Lenguaje de programación orientado a objetos.

**ALGOL 68:** Algorithmic language 1968. Lenguaje de programación de la familia ALGOL.

**ANSI:** Instituto Nacional Americano de Estándares.

**ANSI C:** Estándar publicado por el ANSI.

**ANSI C++:** Estándar publicado por el ANSI.

**AT&T:** American Telephone and Telegraph Corporation (Corporación estadounidense de teléfono y telégrafo).

### B

**BACK-END:** Parte que procesa la entrada desde el front-end.

**BCPL:** Lenguaje de programación combinado básico.

### C

**CASE:** Ingeniería Asistida por Computadora.

**CLR:** Common Language Runtime.

**COM:** Component Object Model. Es una manera neutral de implementar objetos con respecto al lenguaje, de manera que pueden ser usados en multitud de entornos.

**C ++ :** Lenguaje de programación.

D

**DFD:** Diagrama de flujo de datos.

**DLL:** Dynamic Link Library. Biblioteca de enlaces dinámicos. Son rutinas ejecutables disponibles para aplicaciones en tiempo de ejecución.

E

**ECMA:** European Computer Manufacturers Association. Organización internacional dedicada a los estándares para la comunicación y la información.

F

**FCL:** FrameWork Class Library.

**FRONT-END:** Parte del software que interactúa con el usuario.

## G

**GB:** Gigabyte. Unidad de almacenamiento de información equivalente a  $10^9$  bytes.

**GHz:** Procede de Gigahertz y es una medida usada normalmente para determinar la velocidad del procesador.

## H

**HTML:** HyperText Markup Language (Lenguaje de Marcado de Hipertexto). Es el lenguaje de marcado predominante para la elaboración de páginas web.

## I

**IDE:** Integrated Development Environment. Entorno de desarrollo integrado.

**Ingeniería del software:** Disciplina o área de la informática o ciencias de la computación, que ofrece métodos y técnicas para desarrollar y mantener software de calidad que resuelven problemas de todo tipo.

**ISO:** Organización Internacional de Estándares.

**I-CASE:** Integrated CASE, CASE integrado.

## L

**LINQ:** Language Integrated Query. Conjunto de características presentado en Visual Studio 2008 que agrega capacidades de consulta eficaces a la sintaxis de los lenguajes C# y Visual Basic.

**LISTBOX:** Componente dinámico (no tiene tamaño definido) que permite procesar visualmente un conjunto de elementos de tipo string.

**L-CASE:** Lower CASE, CASE inferior.

## M

**MDI:** Multiple Document Interface. Interfaz de múltiples documentos.

**Métrica v3:** Métrica versión 3.

**MFC:** Microsoft Foundation Classes. Conjunto de clases que provee un acceso más sencillo a las API de Windows.

## O

**ODBC:** Open DataBase Connectivity. Estándar de acceso a bases de datos desarrollado por SQL Access Group en 1992

## R

**RAM:** Random Access Memory. Memoria de acceso aleatorio donde la computadora guarda información para que pueda ser procesada más rápidamente. Se almacena toda información que está siendo usada en el momento.

## S

**SERVICE PACK:** Agrupación de actualizaciones, reparaciones y ampliaciones para una aplicación o sistema operativo específico, contenido en un solo paquete ejecutable.

**SIMULA 67:** Fue el primer lenguaje de programación que incorporó el concepto de clase.

**SOAP:** Simple Object Access Protocol. Protocolo basado en XML utilizado para poder comunicarse e interoperar con los servicios web.

**SQL:** Structured Query Language. Es un lenguaje de base de datos normalizado.

**SSADM:** Structured System Analysis and Design Method.

**STL:** Standard Template Library. Es una colección de estructuras de datos genéricas y algoritmos escritos en C++.

**STORED PROCEDURE:** Procedimiento almacenado físicamente en una base de datos.

T

**TIGGERS:** Disparador que define una acción que la base de datos debe llevar a cabo cuando se produce algún suceso relacionado con la misma.

**TOP-DOWN:** Arriba-abajo.

U

**UNIX:** Sistema operativo.

**U-CASE:** Upper CASE, CASE superior.

V

**VSTO:** Visual Studio Tools for Office.

W

**WCF:** Windows Communication Foundation. Modelo de programación unificado de Microsoft para generar aplicaciones orientadas a servicios.

**WPF:** Windows Presentation Foundation. Permite el desarrollo de interfaces de interacción en Windows tomando las mejores características de las aplicaciones Windows y de las aplicaciones web.



**XML:** eXtensible Markup Language. Es un lenguaje de etiquetado extensible para el intercambio de gran variedad de datos.

**XPATH:** XML Path Language. Es un lenguaje que permite construir expresiones que recorren y procesan un documento XML.

## 11. Referencias

Para la realización de este proyecto, se han utilizado algunas referencias bibliográficas, pero la fuente principal de información ha sido internet. Su cita en el texto se hace en la introducción de los capítulos.

Como referencias bibliográficas más importantes cabe destacar:

[ACE+03] Antonio de Amescua Seco, Juan José Cuadrado Gallego, Emilio Ernica Lafuente, Javier García Guzmán, Luis García Sánchez, Paloma Martínez Fernández y M<sup>a</sup> Isabel Sánchez Segura. *Análisis y diseño estructurado y orientado a objetos de sistemas informáticos*. (2003, Madrid).ISBN:8448139240

[Bag02] Bagnall, Brian. *C# for Java programmers*. (2002, Rockland, Massachusetts). ISBN: 9781931836548.

[Ceb07] Ceballos Sierra, Francisco Javier. *Enciclopedia de Microsoft Visual C#*. (2<sup>a</sup>, 2007, Paracuellos del Jarama, Madrid). ISBN: 9788478978106.

[Dei07] Deitel, Harvey M. *Cómo programar en C#*. (2007, Naucalpan de Juárez (México)). ISBN: 9789702610564.

[HC] *Herramientas case*. Instituto nacional de estadística e informática. (Noviembre de 1999).

[Pet06] Petzold, Charles. *.NET Book Zero*. (Diciembre de 2006).

[San94] Sánchez Crespo, José Manuel. *EasyCASE for Microsoft Windows*. (1994, Madrid).

[Sol08] Solis, Daniel Martin. *Illustrated C# 2008*. (2008, Berkeley, California). ISBN: 9781590599549.

[Var07] Vargas C., Aidee. *Análisis de Sistemas*. (Octubre de 2007).

Como portales de internet más importantes de donde se ha extraído información tenemos:



- [1][http://administracionelectronica.gob.es/?\\_nfpb=true&\\_pageLabel=P60085901274201580632&langPae=es](http://administracionelectronica.gob.es/?_nfpb=true&_pageLabel=P60085901274201580632&langPae=es), accedido en 2010.
- [2][http://www.eici.ucm.cl/Academicos/R\\_Villarroel/descargas/ing\\_sw\\_1/Metodologias.pdf](http://www.eici.ucm.cl/Academicos/R_Villarroel/descargas/ing_sw_1/Metodologias.pdf), accedido en 2010.
- [3]<http://www.elguruprogramador.com.ar/foros/c-sharp>, accedido en 2010.
- [4]<http://www.latindevelopers.com/forum/c-csharp-f34.html>, accedido en 2010.
- [5][http://www.lawebdelprogramador.com/foros/C\\_sharp/index1.html#](http://www.lawebdelprogramador.com/foros/C_sharp/index1.html#), accedido en 2010.
- [6]<http://msdn.microsoft.com/library/ms123401>, accedido en 2010.
- [7]<http://msdn2.microsoft.com/en-us/netframework/aa497270.aspx>, accedido en 2010.
- [8][http://www.mygnet.net/foros/c\\_sharp\\_c\\_sharp/](http://www.mygnet.net/foros/c_sharp_c_sharp/), accedido en 2010.
- [9]<http://office.microsoft.com/es-es/visio-help/>, accedido en 2010.
- [10][www.scribd.com/doc/9018328/Historia-C](http://www.scribd.com/doc/9018328/Historia-C), accedido en 2010.
- [11][www.wikipedia.org](http://www.wikipedia.org), accedido en 2010.

La gran mayoría de los portales de internet accedidos no han sido referenciados en el texto del presente proyecto, pero fueron de gran ayuda para el desarrollo del aplicativo, por ello se cree justo que aparezcan.